Environnement Canada Environment Canada

Institut national de recherches en hydrologie

ÉTUDE N° 125, SÉRIE SCIENTIFIQUE DE LA DGEI
RAPPORT INRH N° 9

Programme interactif LAYOUT, conçu pour la mise en page de la documentation relative aux fichiers informatisés imprimés en direct sur des terminaux éloignés

W.C. Horwood et K.U. Weyer

INRH

INSTITUT NATIONAL DE RECHERCHES EN HYDROLOGIE
DIRECTION GÉNÉRALE DES EAUX INTÉRIEURES
CALGARY (ALBERTA), 1982

© Ministre des Approvisionnements et Services Canada 1983

N° de cat. En 36-502/125F

ISBN 0-662-91979-3

Table des matières

Page

RÉSUMÉ v

ABSTRACT v

INTRODUCTION 1

DESCRIPTION GÉNÉRALE 1

EXIGENCES RELATIVES À L'ENTRÉE 2

FONCTIONNEMENT DU PROGRAMME LAYOUT 2
- Procédure PLAY 3
- Instructions introduites au terminal 4
- Identification des caractères de commande de l'imprimante 6
- Structuration de fichiers au moyen de signaux insérés lors de la mise en forme du fichier 7
- Types de fichiers à structurer 7
  - Fichier de programme en images de cartes 7
  - Fichier de données en images de cartes 7
  - Listage de la compilation d'un programme 7
  - Fichier qui contient les résultats d'un programme 8
  - Fichier binaire 8

IMPRESSION D'UNE SORTIE STRUCTURÉE À PARTIR DES FICHIERS D'UNE MÉMOIRE DE MASSE 8

REMERCIEMENTS 8

ANNEXE A. Version non structurée du programme LAYOUT 9

ANNEXE B. Version non structurée d'un tableau de données 21

ANNEXE C. Version structurée du programme LAYOUT 29

ANNEXE D. Version structurée d'un tableau de données 45

ANNEXE E. Procédure PLAY 57

Tableaux

1. Hiérarchie et fonction des sous-programmes employés dans le programme LAYOUT 2
2. Paramètres de mise en page employés avec l'instruction «CHANGE» 3
3. Messages affichés par la procédure PLAY 4

Tableaux (suite)

Page

4. Réaction à l'instruction «HELP» qui fait imprimer les instructions à suivre pour faire fonctionner le programme LAYOUT 4
5. Réaction à l'instruction «LIST»; l'exemple donne la liste des valeurs par défaut des dix paramètres 5
6. Symboles qui représentent la réaction du système d'exploitation NOS 1.0.0 employé avec l'ordinateur CYBER 172. 5
7. Texte affiché lors d'une exécution du programme LAYOUT, quand les valeurs par défaut ont été utilisées pour tous les paramètres. 6
8. Messages d'erreur pour le programme LAYOUT 6
9. Caractères de commande de l'imprimante et leur effet 7

Résumé

Le processus de documentation des listages et sorties des programmes en vue de la production et de la publication des rapports est une tâche longue et fastidieuse, en plus d'être une source d'erreurs. Il est donc préférable que cette opération soit exécutée automatiquement par l'ordinateur qui utilise alors les fichiers mémorisés. Cependant, ces derniers sont le plus souvent stockés comme une suite continue d'informations. Leur impression au terminal produit des listages continus et non structurés dont la lecture est difficile.

Sur le plan théorique, la structure définitive des listages et la disposition de la surface d'impression peuvent être prévues au moment de la programmation. Cependant, il s'agit d'un travail fort long et fort complexe.

Il est possible de contourner tous ces problèmes efficacement en utilisant un programme universel qui structure automatiquement, en vue de l'impression par l'ordinateur, les listages et les entrées et sorties des programmes sur fichier magnétique. À la connaissance des auteurs, le programme LAYOUT est le premier programme général de ce genre. La conception de la présentation des pages en vue de l'impression finale est effectuée de façon interactive aux terminaux à distance. L'impression automatique du fichier structuré définitif est également effectuée aux terminaux à distance afin d'éliminer certaines difficultés inhérentes aux imprimantes principales, comme la réponse automatique aux commandes de mise en forme qui sont incluses dans les sorties du programme.

Abstract

The documentation of program listings and program outputs for reports and publications is a tedious and time-consuming procedure subject to errors. It is best done automatically using the files and the facilities of the computer itself. Computer files, however, are generally stored as continuous streams of information. Printing of these files at terminals produces continuous and unstructured listings, which are very laborious to read.

In theory, the final structure of the listings and the layout of the printing faces can be fully anticipated at the time of programming. This, however, has proven to be a complicated, cumbersome and time-consuming procedure.

All of these disadvantages can be efficiently overcome by using a general purpose program to structure automatically for printing by the computer any computer listing, program input or output stored on magnetic files. To the best knowledge of the authors, program LAYOUT is the first general program of this kind. The design of the page layout for the final printing is prepared interactively at a remote terminal. Automatic printing of the final structured file is also routed to remote terminals to avoid some of the disadvantages of main printers of computers such as their automatic response to formatting commands often included in program output.

Programme interactif LAYOUT, conçu pour la mise en page de la documentation relative aux fichiers informatisés imprimés en direct sur des terminaux éloignés

W. C. Horwood et K. U. Weyer

INTRODUCTION

La documentation relative à un programme d'ordinateur comprend ordinairement des listes distinctes pour le programme lui-même, pour les fichiers de données en entrée et pour les résultats à la sortie. La façon la plus fiable de produire ces listes consiste à faire imprimer les fichiers originaux par l'ordinateur lui-même. En général, ces fichiers ont été créés et mémorisés sur disques sous forme de trains de renseignements continus. Quand on fait imprimer un tel fichier sur un terminal éloigné, la liste est continue et non structurée car les caractères de commande du chariot ne sont généralement pas reconnus. Ce type de liste est plutôt difficile à lire et, en général, la liste ne peut être reproduite telle quelle dans les rapports.

Afin d'illustrer cette situation, des exemples d'imprimés non structurés sont donnés à l'Annexe A dans le cas d'un fichier qui contient un listage du programme LAYOUT, décrit dans le présent document, et à l'Annexe B dans le cas d'un fichier qui contient un tableau renfermant des données produites par suite des calculs effectués par un programme différent. Dans l'Annexe A, l'enregistrement des renseignements est continu, sans tenir compte des limites des feuillets. Il arrive souvent que des lignes d'impression se trouvent très près des perforations du papier. Aucune fonction ne permet de numéroter les pages ou de produire des marges; les sous-programmes se suivent sans être séparés par des lignes vierges. À la fin du fichier, l'ordinateur produit des résultats étrangers à l'état désiré. Dans l'Annexe B, les tableaux paraissent également sous forme de trains de renseignements continus. En plus des imperfections mentionnées pour l'Annexe A, le caractère «1» de commande du chariot paraît à gauche des en-têtes des sous-tableaux. La ligne qui contient ce caractère de commande est décalée vers la droite.

La documentation qui constitue les Annexes C et D a été structurée et préparée au moyen du programme LAYOUT. Les améliorations produites par ce programme sont évidentes quand on compare l'Annexe C à l'Annexe A et l'Annexe D à l'Annexe B.

Au cours de la revue du présent manuscrit, nous avons fait l'essai du programme LAYOUT au moyen d'un terminal éloigné Texas Instruments Silent 700 relié à un ordinateur CDC (Control Data Corporation) CYBER 74 qui utilisait le système d'exploitation NOS/BE 1.2 niveau 447. Ce système élimine automatiquement le caractère «1» de commande du chariot quand on utilise «COPYCR» pour effectuer l'impression d'un fichier sortie sur un terminal éloigné. Selon notre expérience, il semble qu'il s'agisse d'une exception, qui pourrait dépendre d'une caractéristique du logiciel utilisé avec le terminal et du téléimprimeur employé. Habituellement, les fichiers de sortie produits sur un terminal éloigné sont imprimés de façon non structurée tel qu'indiqué dans l'Annexe B. Nous recommandons, de toute façon, de ne pas utiliser le caractère de commande du chariot pour imprimer des fichiers sortie sur des systèmes de cette nature, quand on compte structurer ces fichiers selon les dix options mentionnées plus loin.

Afin d'améliorer la documentation, le programme LAYOUT a été écrit, de façon à structurer, de manière interactive, la présentation des pages pour tout fichier qui doit être imprimé sur un terminal éloigné. Il a été conçu de façon à pouvoir être utilisé pour imprimer la documentation relative à des fichiers informatisés au moyen d'ordinateurs qui disposent de fonctions de traitement interactif. Le programme devrait donc intéresser de façon générale tout utilisateur des ordinateurs à partir de terminaux éloignés. Pour tirer pleinement profit de certaines fonctions du programme, il faut pouvoir disposer d'un terminal éloigné qui peut imprimer 137 caractères par ligne. Le programme est facile à employer. L'utilisateur peut, au cours de l'exécution du programme, employer toutes les instructions d'exploitation dont il peut avoir besoin après l'introduction d'une instruction déterminée.

DESCRIPTION GÉNÉRALE

Le programme LAYOUT peut être utilisé pour structurer une page afin qu'elle soit imprimée par un terminal éloigné; il permet à l'utilisateur:

(1) de déterminer les dimensions des marges supérieures et inférieures;

(2) de déterminer les dimensions des marges de gauche et de droite;
(3) d'ajouter la pagination pour l'impression finale;
(4) d'éliminer certains renseignements qui seraient normalement imprimés à la fin des lignes, par exemple les codes d'identification qui figurent dans les colonnes 73-80 des listages des programmes;
(5) d'insérer des lignes vierges entre les sous-programmes;
(6) de positionner un sous-programme au haut d'une nouvelle page;
(7) de structurer les tableaux ou d'autres fichiers à imprimer au moyen du caractère «1» de commande du chariot ou de l'instruction «CEND»;
(8) d'éliminer l'impression du caractère «1» de commande du chariot et de l'instruction «CEND»;
(9) d'annuler la distorsion des lignes causée par l'insertion du caractère «1» de commande du chariot dans la sortie sur imprimante;
(10) d'éliminer les messages produits par l'ordinateur après la fin d'un fichier.

Le programme LAYOUT a été écrit en FORTRAN Extended Version 4 pour les ordinateurs CDC qui utilisent l'un des systèmes d'exploitation NOS ou KRONOS. Les auteurs en ont fait l'essai au moyen d'un terminal DTC (Data Terminals and Communication) 300/S relié par un coupleur acoustique et une ligne téléphonique à l'ordinateur CDC CYBER 172 de l'Université de Calgary qui utilise le système d'exploitation NOS 1.0.0 niveau 411.

Tableau 1. Hiérarchie et fonction des sous-programmes employés avec le programme LAYOUT

Hiérarchie des sous-programmes	Fonctions
LAYOUT	Programme principal.
RDCMND	Lit les instructions et appelle les sous-programmes appropriés.
LIST	Produit la liste des valeurs courantes des paramètres.
CHANGE	Modifie les valeurs des paramètres.
ERROR	Imprime les messages d'erreur.
HELP	Imprime les instructions à suivre pour faire fonctionner le programme.
NEWPAGE	Positionne la surface d'impression sur une nouvelle page et imprime le numéro de la page.
ENDCHK	Repère la fin des parties du programme.

Nota: Le décalage indique d'où proviennent les appels aux sous-programmes; LIST, ERROR et HELP peuvent aussi être appelés au cours du déroulement du sous-programme CHANGE.

Lors des essais sur l'ordinateur CDC CYBER 74 utilisant le système d'exploitation NOS/BE 1.2 niveau 447, nous avons découvert qu'il fallait modifier de façon significative la procédure PLAY (qui commande le fonctionnement du programme LAYOUT), afin de tenir compte des caractéristiques logiques différentes du système d'exploitation NOS employé. Cependant, ce rapport décrira, à titre d'exemple, la procédure PLAY telle que les auteurs l'ont utilisée.

Le programme occupe $27\,000_8$ mots (octal). La hiérarchie et les fonctions des sous-programmes utilisés dans le programme LAYOUT sont décrites dans le tableau 1.

EXIGENCES RELATIVES À L'ENTRÉE

Avant d'utiliser le programme LAYOUT, il faut disposer des renseignements suivants:

(1) le type de fichier à structurer;
(2) le nombre maximum de lignes qui peuvent être imprimées sur chaque page par un terminal déterminé;
(3) le nombre maximum de caractères qui peuvent être imprimés sur une ligne par ce terminal.

Pour l'exécution interactive du programme LAYOUT, il faut pouvoir accéder aux trois fichiers suivants dans la mémoire de masse en direct de l'ordinateur:

(1) une version en images de cartes ou binaire du programme LAYOUT;
(2) le fichier de la procédure PLAY afin de pouvoir compiler et exécuter le programme;
(3) le fichier qui doit être structuré par le programme LAYOUT.

FONCTIONNEMENT DU PROGRAMME LAYOUT

L'utilisateur détermine la structure de la page en définissant les dix paramètres indiqués au tableau 2. Ces paramètres possèdent des valeurs par défaut affectées par le programme; elles sont indiquées dans le tableau 2. Cependant, l'utilisateur peut modifier toutes ces valeurs, de façon interactive, au moment de l'exécution du programme. Une fois les valeurs des paramètres affectées et l'instruction «PROCEED» introduite, le programme traite le fichier entrée à structurer selon les dix paramètres de mise en page. Un fichier sortie structuré est alors créé. Ce fichier peut ensuite être imprimé au terminal au moyen des procédures indépendantes du programme décrites à la rubrique «Impression d'une sortie structurée à partir des fichiers d'une mémoire de masse».

Tableau 2. Paramètres de mise en page employés avec l'instruction «CHANGE»

Ordre	Paramètre	Valeurs permises	Valeur par défaut	Description
1	CHKFNP	0,1,2,3	3	Le programme repère le début d'une nouvelle page ou la fin d'un sous-programme; 0 = aucune opération de repérage; 1,2 = le programme repère la présence du caractère «1» de contrôle du chariot, élimine ce caractère de l'impression et décale la ligne d'une colonne vers la gauche; 1 = début d'une nouvelle page, 2 = ne pas commencer une nouvelle page; 3 = le programme repère les instructions «END» dans les colonnes 7 à 72.
2	PGNUM	0,1	1	Option pour la numérotation des pages: 0 = aucun numéro de page imprimé, 1 = impression des numéros de page.
3	NCOL	1-137	72	Nombre maximal de colonnes à imprimer sur chaque ligne.
4	LFTMG	1-136	10	Largeur de la marge de gauche, en colonnes. Note: NCOL + LFTMG ⩽ 137
5	MXLINE	1-99	66	Nombre maximal de lignes imprimées, au terminal, sur chaque page entre les perforations. Note: MXLINE = TOPMG + BOTMG + NLPRT
6	TOPMG	0, MXLINE-1	6	Nombre de lignes qui constituent la marge supérieure.
7	BOTMG	0, MXLINE-1	5	Nombre de lignes qui constituent la marge inférieure.
8	NLPRT	1, MXLINE	55	Nombre de lignes imprimées par page.
9	DIVSPC	0-MXLINE	6	Nombre de lignes entre la fin d'un sous-programme et le début du sous-programme suivant.
10	MNLPG	0-MXLINE	9	Nombre minimum de lignes qui doivent pouvoir être imprimées sur une page avant de commencer l'impression d'un nouveau sous-programme sur la page en cours d'impression.

Les exigences et les conditions relatives à l'exploitation sont décrites en détail dans les sous-sections suivantes qui traitent de:

(1) la procédure PLAY, un listage d'instructions dans le langage des ordres de gestion du système d'exploitation NOS 1.0.0 pour un ordinateur CDC CYBER 172 qui servent à commander l'exécution du programme (cette procédure est donnée comme exemple);
(2) les instructions introduites au terminal afin de dialoguer avec le programme LAYOUT;
(3) les caractères de commande de l'imprimante identifiés par le programme LAYOUT;
(4) la structuration des états par la mise en forme des fichiers;
(5) les types de fichiers qui doivent être structurés par le programme LAYOUT.

Procédure PLAY

La procédure PLAY (*P*rocédure pour le programme *LAY*OUT) est un moyen commode de traiter les suites d'instructions ainsi que l'affectation des noms de fichiers. Elle contient les ordres de gestion qui servent à compiler et à exécuter le programme LAYOUT. Cette procédure a été écrite pour l'ordinateur CDC CYBER 172 de l'Université de Calgary qui utilise le système d'exploitation NOS 1.0.0 niveau 411. Pour les autres systèmes d'exploitation de CDC tels KRONOS ou NOS/BE 1.2 niveau 447, ou pour les autres systèmes informatiques, il faut écrire différents fichiers de procédures afin de tenir compte des différences dans le logiciel et dans le matériel. Cependant, afin de faciliter l'élaboration de procédures semblables pour d'autres systèmes d'exploitation, nous avons inclus dans l'Annexe E une analyse détaillée de la structure et de l'utilisation de la procédure PLAY pour le système d'exploitation NOS 1.0.0.

Au cours de son exécution, la procédure PLAY communique certains messages à l'utilisateur; on en trouve la liste dans le tableau 3. Certains de ces messages signalent la fin de certaines phases de l'exécution du passage-machine, tandis que d'autres sont produits à la suite d'erreurs. L'exécution de la procédure PLAY se termine quand une erreur est détectée.

Instructions introduites au terminal

Il se peut que l'utilisateur désire modifier les valeurs par défaut des dix paramètres utilisés pour définir la surface d'impression. On peut utiliser quatre instructions à cette fin.

«HELP» cette instruction décrit les quatre instructions ainsi que les paramètres utilisés pour l'instruction «CHANGE». Le programme réagit à l'instruction «HELP» en imprimant des instructions qui permettent de faire fonctionner le programme; on en trouve la liste au tableau 4.

Tableau 3. Messages affichés par la procédure PLAY

Message	Motif
COMMENT. MISSING INPUT FILE. (Commentaire. Le fichier entrée manque)	Erreur détectée.
COMMENT. MISSING OUTPUT FILE NAME. (Commentaire. Le nom du fichier sortie manque)	Erreur détectée.
COMMENT. MISSING PROGRAM FILE. (Commentaire. Le fichier du programme manque)	Erreur détectée.
COMMENT. PROGRAM HAS BEEN COMPILED. (Commentaire. Le programme a été compilé)	Messages qui signalent la fin de certaines phases de l'exécution du passage-machine.
COMMENT. COMPILER LISTING HAS BEEN SAVED. (Commentaire. Le listage de la compilation a été préservé)	
COMMENT. BINARY PROGRAM HAS BEEN SAVED. (Commentaire. Le programme en binaire a été conservé)	
COMMENT. BINARY PROGRAM IS AVAILABLE. (Commentaire. Le programme binaire est disponible)	
COMMENT. OUTPUT FILE HAS BEEN CREATED AND SAVED. (Commentaire. Le fichier sortie a été créé et conservé)	
COMMENT. *** END OF RUN *** (Commentaire. *** fin du passage-machine ***)	

Remarque: Le terme «COMMENT» indique que ces messages sont produits par la procédure PLAY.

Tableau 4. Réaction à l'instruction «HELP» qui fait imprimer les instructions à suivre pour faire fonctionner le programme LAYOUT

```
COMMANDS FOR LAYOUT:
A. HELP    - LISTS "LAYOUT" COMMANDS AND PARAMETERS
B. LIST    - LISTS CURRENT PARAMETER VALUES
C. CHANGE  - ALLOWS CURRENT PARAMETER VALUE(S) TO BE CHANGED.
           - NEW VALUES MUST BE ENTERED IN THE FOLLOWING ORDER:
               1. CHKFNP  - NEW PAGE CHECK (0=NO CHECK, 1="1" ON,
                            2="1" OFF, 3="END")
               2. PGNUM   - PAGE NUMBER (1=YES,0=NO)
               3. NCOL    - NUMBER OF COLUMNS PER LINE
               4. LFTMG   - LEFT MARGIN (IN COLUMNS)
               5. MXLINE  - MAX. NUMBER OF PRINT LINES PER PAGE
               6. TOPMG   - TOP MARGIN (IN LINES)
               7. BOTMG   - BOTTOM MARGIN (IN LINES)
               8. NLPRT   - NUMBER OF LINES PRINTED PER PAGE
               9. DIVSPC  - SPACE BETWEEN SUBPROGRAMS (IN LINES)
              10. MNLPG   - MIN. NO. OF LINES TO START SUBPROGRAM ON
                            CURRENT PAGE
           - TO RETAIN CURRENT PARAMETER VALUE, ENTER A STAR FOR THE
             CORRESPONDING PARAMETER IF IT PRECEDES A NEW VALUE.
             FOR EXAMPLE, TO CHANGE THE VALUE OF NCOL TO 80,
                 ENTER:*,*,80
           - COMMANDS "HELP" AND "LIST" MAY BE ADDRESSED FROM WITHIN
             THE "CHANGE" MODE.
           - TO EXIT FROM THE COMMAND "CHANGE" WITHOUT ENTERING ANY NEW
             VALUES, ENTER A STAR
D. PROCEED- IS THE LAST COMMAND TO BE ENTERED; PARAMETER VALUES ARE
            SATISFACTORY, STRUCTURING OF INPUT FILE WILL PROCEED
```

«LIST» cette instruction donne la liste des valeurs des paramètres, dans l'ordre précisé dans le tableau 5. La signification de ces paramètres est expliquée au tableau 2.

Tableau 5. Réaction à l'instruction «LIST»; l'exemple donne la liste des valeurs par défaut des dix paramètres

```
LIST OF CURRENT PARAMETER VALUES FOR LAYOUT
      1. CHKFNP  =  3
      2. PGNUM   =  1
      3. NCOL    = 72 COLUMNS
      4. LFTMG   = 10 COLUMNS
      5. MXLINE  = 66 LINES
      6. TOPMG   =  6 LINES
      7. BOTMG   =  5 LINES
      8. NLPRT   = 55 LINES
      9. DIVSPC  =  6 LINES
     10. MNLPG   =  9 LINES
```

«CHANGE» cette instruction permet à l'utilisateur d'introduire de nouvelles valeurs pour les paramètres, tels qu'expliqués dans le tableau 2. Le terminal réagit à l'introduction de cette instruction en imprimant le message «ENTER NEW LAYOUT PARAMETERS:» (introduire de nouveaux paramètres pour la présentation) suivi d'un point d'interrogation sur la ligne suivante ou d'un autre type de caractère de sollicitation selon le système employé. On trouve dans le tableau 6, la liste de tous les symboles employés dans le présent document, pour préciser, la réaction de l'ordinateur.

Les nouvelles valeurs précisées à la suite de l'introduction de l'instruction «CHANGE» sont introduites sous forme d'entiers séparés par des virgules. Ces valeurs doivent être introduites dans l'ordre précisé dans les tableaux 2 et 5. On peut employer des astérisques quand on désire conserver les valeurs courantes des paramètres plutôt que d'introduire les valeurs de ces paramètres. Si, par exemple, on désire donner au paramètre 3, NCOL, une nouvelle valeur, 80, et si on veut conserver les valeurs courantes de CHKFNP et de PGNUM, voici quelle doit être la présentation de la ligne introduite au terminal:

*, *, 80 ou
* * 80 ou
3, 1, 80

Il n'est pas nécessaire d'utiliser des astérisques pour conserver les valeurs courantes des paramètres qui suivant NCOL.

Lorsqu'il reçoit les paramètres, le programme remplace les anciennes valeurs par les nouvelles et, par la suite, quitte le mode «CHANGE». Pour quitter le mode «CHANGE» sans modifier les valeurs des paramètres, il suffit d'introduire un astérisque ou toute combinaison de valeurs courantes.

L'utilisateur peut accéder aux instructions «LIST» et «HELP» quand le programme est en mode «CHANGE», s'il a besoin d'aide. Une fois l'aide apportée, le programme reste en mode «CHANGE».

«PROCEED» c'est la dernière instruction qui doit être introduite. À ce moment, les valeurs des paramètres ont été affectées et le programme commence la structuration du fichier entrée. Quand on utilise les valeurs par défaut pour tous les paramètres de mise en page, on peut faire exécuter le programme en n'utilisant que l'instruction «PROCEED». On trouve au tableau 7 un exemple de ce genre d'exécution du programme.

Toutes les instructions et toutes les valeurs des paramètres sont transférées du terminal à l'ordinateur quand on appuie sur la touche de retour du chariot. Si une instruction n'est pas introduite correctement ou si

Tableau 6. Symboles qui représentent la réaction du système d'exploitation NOS 1.0.0 employé avec l'ordinateur CYBER 172

Reaction au terminal	Origine	Signification	Mesures à prendre
/	Le système d'exploitation NOS est en mode de traitement en temps partagé ou par lots	Introduction d'une ou plusieurs commandes de traitement par lots	Appeler la procédure PLAY ou commencer à reproduire un fichier structuré.
?	Programme LAYOUT	Le programme demande des données	Introduire une instruction du programme LAYOUT ou les paramètres relatifs à l'instruction «CHANGE».
*	Procédure PLAY	Fin de la procédure PLAY	Aucune mesure à prendre, car l'astérisque sera suivi d'une barre oblique.

Tableau 7. Texte affiché lors d'une exécution du programme LAYOUT, quand les valeurs par défaut ont été utilisées pour tous les paramètres

```
CALL,PLAY(BINARY=BINLAY,INFILE=FILE2,OUTFILE=FILE3)
COMMENT. BINARY PROGRAM IS AVAILABLE.

*****************************   L A Y O U T   *****************************
  ENTER LAYOUT COMMANDS:
  ----------------------
? PROCEED

***************   PROGRAM PROCEEDS WITH FILE STRUCTURING   *************

COMMENT. OUTPUT FILE HAS BEEN CREATED AND SAVED.
COMMENT.            *** END OF RUN ***
*
/
```

Remarque: L'utilisateur a transmis la première ligne au système d'exploitation NOS. Le dialogue avec le programme LAYOUT est compris entre les deux lignes d'astérisques. La barre oblique a été produite par le système d'exploitation NOS. Tous les autres messages ont été produits par la procédure PLAY. Pour connaître la signification de ?, * et / voir le tableau 6.

Tableau 8. Messages d'erreur pour le programme LAYOUT

Message*	Explication
ILLEGAL COMMAND, FOR ASSISTANCE ENTER "HELP" (instruction interdite; pour obtenir de l'aide introduire «HELP»)	Une instruction interdite a été introduite; la liste de toutes les instructions valables est produite par l'instruction «HELP».
ILLEGAL CHARACTER IN PARAMETER LIST FOR "CHANGE" COMMAND (caractère interdit ans la liste des paramètres pour l'instruction «CHANGE»)	Les valeurs des paramètres doivent être des entiers positifs, séparés par des espaces ou des virgules. Voici la liste des caractères valides: astérisques, espaces, virgules et chiffres.
MXLINE MUST EQUAL TOPMG + BOTMG + NLPRT. (MXLINE doit être égal à TOPMG + BOTMG + NLPRT)	TOPMG + BOTMG + NLPRT n'est pas égal à MXLINE; vérifier les valeurs de ces quatre paramètres.
NCOL + LFTMG MUST BE LESS THAN OR EQUAL TO 137 (NCOL + LFTMG doit être inférieur ou égal à 137)	NCOL + LFTMG est supérieur à 137; vérifier les valeurs de NCOL et LFTMG.
VALUE FOR PARAMETER X IS OUT OF RANGE (la valeur du paramètre X ne fait pas partie de la gamme de valeurs permises)	Vérifier la valeur du $X^{ième}$ paramètre (X = 1,2. . .10). On trouve au tableau 2 les gammes de valeurs permises pour chaque paramètre.

*Chaque message est précédé de: ***ERROR*** (***erreur***).

la valeur d'un paramètre n'est pas comprise dans la gamme de valeurs permises, le programme fait imprimer un message d'erreur. Pour corriger une erreur, il suffit de réintroduire toute la ligne et de la transmettre. La liste de tous les messages d'erreur du programme LAYOUT ainsi que les explications pertinentes se trouvent au tableau 8.

Identification des caractères de commande de l'imprimante

La première colonne de chaque ligne qui doit être produite en sortie sur l'imprimante est utilisée pour la commande du chariot. Les imprimantes par ligne des centres de calcul éliminent automatiquement les caractères de commande du chariot lors de l'impression; mais tel n'est pas le cas pour la majorité des terminaux à distance.

Les caractères FORTRAN de commande de l'imprimante employés pour la plupart des ordinateurs sont indiqués au tableau 9. Les caractères de commande de l'imprimante utilisés en plus de ceux qui sont mentionnés au tableau 9 dépendent habituellement du système employé; ils sont écrits dans le guide de référence du système d'exploitation utilisé. La présente version du programme LAYOUT réagit aux caractères de commande espace et «1». Il est possible de modifier le programme quand on doit utiliser d'autres caractères de commande de l'imprimante.

Le paramètre CHKFNP est utilisé pour manipuler le caractère «1» de commande du chariot. Si CHKFNP = 1, le programme élimine le caractère «1» de commande du chariot et décale la ligne d'une colonne vers la gauche; il

insère aussi le nombre approprié de lignes vierges afin de sauter à la page suivante avant de recommencer l'impression. Si CHKFNP = 2, le programme élimine aussi le caractère «1» de commande du chariot et décale la ligne d'une colonne vers la gauche, mais il n'insère par de ligne vierge. Si CHKFNP = 0, le programme saute le caractère «1» de commande du chariot. Cependant, selon le type de terminal et le système informatique utilisés, le caractère de commande du chariot peut servir à effectuer un saut de page sur le terminal ou bien être sauté.

Tableau 9. Caractères de commande de l'imprimante et leur effet

Caractère	Effet
espace	Saut d'une ligne avant l'impression.
0	Saut de deux lignes avant l'impression.
1	Saut à la première ligne de la page suivante avant l'impression.
+	Aucun saut de ligne avant l'impression; ce caractère permet d'effectuer une surimpression.

Structuration de fichiers au moyen de signaux insérés lors de la mise en forme du fichier

Jusqu'ici, nous n'avons étudié que la conception de la surface d'impression, pour des fichiers informatisés, par des méthodes qui n'entraînent pas de modifications du contenu réel de ces fichiers. Cependant, le programme LAYOUT permet aussi d'utiliser des signaux afin de continuer l'impression au haut de la page suivante. Ces signaux doivent être insérés, par une mise en forme, dans le fichier non structuré.

Deux signaux peuvent être employés: un caractère «1» de commande du chariot inséré dans la colonne un de la première ligne de la nouvelle page, ou l'instruction «CENÐ», «C END» ou «C END» entre les deux pages. L'effet de l'insertion du charactère «1» de commande du chariot a déjà été décrit. La présence du signal «CEND» ou d'un des deux autres signaux équivalents donne lieu à une structuration indédpendante. L'impression se poursuit au haut de la page suivante. Tous les autres paramètres continuent à fonctionner.

Il faut insérer les instructions «CEND» de façon qu'elles commencent dans la colonne 1. Quand une ligne contient «CEND», *toutes* les colonnes qui suivent le signal doivent être vierges. Sinon, la ligne n'est pas identifiée comme signal et l'ordinateur la fait imprimer. Si le signal est identifié, la ligne qui contient le signal n'est pas imprimée.

Types de fichiers à structurer

Tout fichier à structurer doit être disponible en direct dans une zone de la mémoire de masse. Un tel fichier possède une des formes suivantes:

Fichier de programme en images de cartes

Ce fichier est une image d'un jeu de cartes constituant un programme. Aucun caractère de commande du chariot ne fait partie de ce fichier. Aucune ligne ne peut contenir plus de 80 caractères, de façon à correspondre à la longueur d'article d'une image de carte. Si le programme est écrit en FORTRAN, le programme peut distinguer les sous-programmes si CHKFNP possède la valeur 3. Le programme LAYOUT repère alors la fin d'un programme en vérifiant si les colonnes 7 à 72 contiennent une instruction «END». On peut alors donner au paramètre DIVSPC une valeur qui permet de laisser, sur la même page, DIVSPC lignes entre la fin d'un sous-programme et le début de celui qui suit. Quand le nombre de lignes disponibles est inférieur au nombre précisé par DIVSPC, le début du sous-programme suivant se trouve au haut de la page suivante. Le paramètre MNLPG positionne aussi le début d'un sous-programme au haut de la page suivante, quand on dispose de moins de MNLPG lignes sur la page en cours d'impression. S'il faut toujours faire commencer l'impression des sous-programmes sur une nouvelle page, il suffit de donner à l'un des paramètres MNLPG ou DIVSPC la valeur de MXLINE.

Fichier de données en images de cartes

Comme c'est le cas pour le fichier de programme en images de cartes, ce fichier constitue une image d'un paquet de cartes. Par conséquent, la valeur du paramètre NCOL ne pourra dépasser 80. De plus, aucun caractère de commande du chariot ne se trouvera dans ce fichier. Afin de s'assurer que les données qui se trouvent dans la colonne 1 ne sont pas identifiées comme des caractères de commande du chariot, il faut donner au paramètre CHKFNP la valeur 0.

Listage de la compilation d'un programme

Comme ce type de fichier a été créé par l'ordinateur sous forme de fichier destiné à l'imprimante par ligne, la valeur du paramètre qui précise le nombre de caractères imprimés par ligne, NCOL, peut aller jusqu'à 137, selon le type d'ordinateur employé. Le listage de la compilation d'un programme FORTRAN comprend habituellement les caractères de commande du chariot, insérés par l'ordinateur, qui signalent le début d'une nouvelle page. Si ces caractères doivent être utilisés pour la structuration, il faut donner au paramètre CHKFNP la valeur 1; le programme LAYOUT repérera alors les caractères «1» de commande du chariot dans la colonne 1.

Fichier qui contient les résultats d'un programme

Ce fichier est un fichier à imprimer créé au cours de l'exécution d'un programme. La valeur du paramètre NCOL pourra donc aller jusqu'à 137. Le fichier peut contenir les caractères de commande du chariot et les numéros de page précisés par le programme qui l'a créé. Les numéros de page contenus dans le fichier ne peuvent être éliminés par le programme LAYOUT. Comme dans le listage de la compilation d'un programme, on peut employer le paramètre CHKFNP pour effectuer la structuration.

Fichier binaire

Un tel fichier contient un ensemble d'instructions traduites du langage de programmation employé pour écrire le programme en langage machine.

IMPRESSION D'UNE SORTIE STRUCTURÉE À PARTIR DES FICHIERS D'UNE MÉMOIRE DE MASSE

La sortie produite par le programme LAYOUT est structurée selon les critères imposés et elle est mémorisée dans une zone de la mémoire de masse de l'ordinateur. La sortie est alors prête à être imprimée sur le terminal éloigné. L'impression de ce fichier peut s'effectuer de diverses façons, qui sont toutes indépendantes du programme LAYOUT. On peut, par exemple, employer un système de mise en forme de texte comme le UCEDIT de l'Université de Calgary pour imprimer ce fichier en tout ou en partie. Il est aussi possible de faire imprimer le fichier sortie au moyen des instructions des ordres de gestion de l'ordinateur. Dans tous les cas, il faut que l'instruction qui ordonne l'impression sur le terminal se trouve au bas d'une page, juste avant le pointillé. De cette façon, on s'assure que le début de la sortie se trouvera au haut de la page suivante. Voici comment serait donnée l'instruction d'impression dans le langage de gestion du système d'exploitation NOS:

```
COPYCR, N5 (N5 = nom du fichier sortie)
----------------------------------------
```

(pointillé entre les pages)

Les systèmes de mise en forme de texte permettent habituellement de commencer l'impression en un point quelconque des fichiers.

Au terminal éloigné, le fichier structuré est imprimé sur du papier en continu non ligné. Une fois ce papier coupé aux dimensions voulues au moyen de l'appareil approprié, il est prêt à servir d'original pour l'impression ou la reproduction.

REMERCIEMENTS

Nous désirons remercier M. G. Grove, d'Ottawa, qui en plus d'effectuer une lecture critique du manuscrit a aussi effectué l'essai du programme dans une installation informatique CDC CYBER 74. Nous désirons aussi remercier M. H.K. Pearse, de Calgary, pour ses commentaires.

Annexe A
Version non structurée du programme LAYOUT

```
C*****************************   L A Y O U T   ***************************C
C                                                                          C
C     MAINLINE PROGRAM TO DESIGN PRINTING FACES FOR DOCUMENTATION          C
C     OF COMPUTER FILES PRINTED ON-LINE AT REMOTE TERMINALS.               C
C                                                                          C
C                          AUTHOR - W.C.HORWOOD                            C
C                                                                          C
C*************************************************************************C
      PROGRAM LAYOUT(INPUT,OUTPUT,A,B,TAPE1=A,TAPE2=B,TAPE5=INPUT,
     *               TAPE6=OUTPUT)
      IMPLICIT INTEGER(D)
      INTEGER TOPMG,BOTMG,PGNUM,DIVSPC,CHKFNP,OUT,ENDFLG
      DIMENSION END(3)
      REAL  MARG(137)
      COMMON /C1000/ CHKFNP, PGNUM, NCOL, LFTMG, MXLINE, TOPMG, BOTMG,
     *              NLPRT, DIVSPC, MNLPG, MARG, NLINE, NSUM1, NSUM2,
     *              MT1, MT2, NPAGE, NCENPG
      COMMON /C2000/ KR, KW, IN, OUT
      COMMON /C3000/ CCARD(80),ICHGCD,ICOL,IPROC
      COMMON /C4000/ CARD(137), ENDFLG, NBCOL
      DATA CC1/1H1/, BLANK/1H /
      DATA END/1HE, 1HN, 1HD/,  C/1HC/
      DATA DTOPMG/6/,   DNLPRT/55/,   DBOTMG/5/,   DMXLINE/66/,
     *     DDIVSPC/6/, DMNLPG/9/,    DPGNUM/1/,   DNCOL /72/,
     *     DLFTMG/10/, DCHKFNP/3/
      DATA KR/1/, KW/2/, IN/5/, OUT/6/
C
      NPAGE=0
      ICHGCD=0
C
C.....INITIALIZE VALUES
      TOPMG=DTOPMG
      BOTMG=DBOTMG
      NCOL =DNCOL
      NLPRT=DNLPRT
      MXLINE=DMXLINE
      DIVSPC=DDIVSPC
      MNLPG=DMNLPG
      PGNUM=DPGNUM
      LFTMG=DLFTMG
      DO 5 I=1,LFTMG
    5 MARG(I)=BLANK
      CHKFNP=DCHKFNP
      NSUM1=TOPMG+NLPRT
      NSUM2=MXLINE-(MNLPG+DIVSPC+BOTMG)
      MT1=TOPMG/2
      MT2=TOPMG-MT1-1
C
      WRITE(OUT,6)
    6 FORMAT(/27(1H*),17H   L A Y O U T   ,28(1H*)/
     * 24H  ENTER LAYOUT COMMANDS:/2X,21(1H-))
   99 CONTINUE
      READ(IN,7) CCARD
    7 FORMAT(80A1)
      CALL RDCMND,RETURNS(99,8)
    8 CONTINUE
      WRITE(OUT,9)
    9 FORMAT(/14(1H*),44H   PROGRAM PROCEEDS WITH FILE STRUCTURING
     * ,14(1H*)/)
C
C
C.....POSITION PAPER TO PAGE ONE, FIRST PRINTED LINE
      NPAGE=NPAGE+1
      NCENPG=NCOL/2+LFTMG
      IF(TOPMG.LE.0) GO TO 11
      DO 10 I=1,TOPMG
```

```
   10 WRITE(KW,1)
    1 FORMAT(1X)
   11 CONTINUE
      NLINE=TOPMG
C
C.....READ NEXT CARD
  100 READ(KR,2) (CARD(I),I=1,NCOL )
    2 FORMAT(137A1)
      IF(EOF(KR)) 999,110
C
  110 CONTINUE
      DTBLNK=NCOL+1
      DO 115 I=1,NCOL
      DTBLNK=DTBLNK-1
      IF(CARD(DTBLNK).NE.BLANK) GO TO 116
  115 CONTINUE
  116 CONTINUE
      NBCOL=DTBLNK
C
C     CHECK FOR "C END" IN COLS. 1 TO 6
      IF(CARD(1).NE.C) GO TO 119
      I=1
  117 I=I+1
      IF(I.GT.4) GO TO 119
      IF(CARD(I).EQ.BLANK) GO TO 117
      IF(CARD(I).NE.END(1).OR.CARD(I+1).NE.END(2).OR.
     *   CARD(I+2).NE.END(3)) GO TO 119
      I1=I+3
      DO 118 I=I1,NBCOL
      IF(CARD(I).NE.BLANK) GO TO 119
  118 CONTINUE
      CALL NEWPAGE,RETURNS(100)
C
  119 CONTINUE
C.....CHECK OPTIONS FOR CHKFNP
      IF(CHKFNP.EQ.0) GO TO 200
      IF(CHKFNP.EQ.3) CALL ENDCHK,RETURNS(100,200)
C     CHKFNP=1 OR 2...CHECK FOR CARRIAGE CONTROL CHARACTERS
      IF(CARD(1).NE.CC1) GO TO 200
C
      DO 130 I=2,NBCOL
  130 CARD(I-1)=CARD(I)
      NBCOL=NBCOL-1
      IF(NBCOL.EQ.0) NBCOL=1
C
      IF(CHKFNP.EQ.1.AND.NLINE.GT.TOPMG) CALL NEWPAGE,RETURNS(200)
C
  200 CONTINUE
      IF(NLINE.GE.NSUM1) CALL NEWPAGE,RETURNS(205)
      IF(NBCOL.EQ.0) GO TO 100
  205 CONTINUE
      IF(LFTMG.GT.0) WRITE(KW,3) (MARG(I),I=1,LFTMG),(CARD(I),I=1,NBCOL)
      IF(LFTMG.LE.0) WRITE(KW,3) (CARD(I),I=1,NBCOL)
      NLINE=NLINE+1
      GO TO 100
    3 FORMAT(137A1)
C
C     SKIP TO NEWPAGE AT END
  999 CONTINUE
      IF(NLINE.GE.(TOPMG+1)) CALL NEWPAGE,RETURNS(1000)
 1000 STOP
      END
C**************************    R D C M N D    **************************C
C                                                                       C
C     THIS SUBROUTINE READS COMMANDS AND CALLS APPROPRIATE              C
C     SUBROUTINES.                                                      C
```

```
C                                                                       C
C***********************************************************************C
      SUBROUTINE RDCMND,RETURNS(M,N)
      DIMENSION CMND(4,4)
      INTEGER TOPMG,BOTMG,PGNUM,DIVSPC,CHKFNP,OUT
      REAL MARG(137)
      COMMON /C1000/ CHKFNP, PGNUM, NCOL, LFTMG, MXLINE, TOPMG, BOTMG,
     *               NLPRT, DIVSPC, MNLPG, MARG, NLINE, NSUM1, NSUM2,
     *               MT1, MT2, NPAGE, NCENPG
      COMMON /C2000/ KR,KW,IN,OUT
      COMMON /C3000/ CCARD(80),ICHGCD,ICOL,IPROC
      DATA CMND/ 1HH, 1HL, 1HC, 1HP,
     *           1HE, 1HI, 1HH, 1HR,
     *           1HL, 1HS, 1HA, 1HO,
     *           1HP, 1HT, 1HN, 1HC /
      DATA BLANK/1H /
C
    1 CONTINUE
      IPROC=0
C     SKIP BLANKS
      ICOL=1
      DO 10 I=1,76
      ICOL=I
      IF(CCARD(I).NE.BLANK) GO TO 20
   10 CONTINUE
      CALL ERROR(1,0),RETURNS(999)
C
C     READ COMMAND
   20 CONTINUE
      DO 30 J=1,4
      ICMND=J
      IF(CCARD(ICOL).EQ.CMND(J,1)) GO TO 40
   30 CONTINUE
      IF(ICHGCD.EQ.1) CALL ERROR(2,0),RETURNS(999)
      CALL ERROR(1,0),RETURNS(999)
C
   40 DO 50 K=2,4
      ICOL=ICOL+1
      IF(CCARD(ICOL).EQ.CMND(ICMND,K)) GO TO 50
C     INVALID COMMAND
      CALL ERROR(1,0),RETURNS(999)
   50 CONTINUE
C
C     VALID COMMAND LOCATED
      GOTO(70,80,90,100),ICMND
   70 CALL HELP,RETURNS(999)
   80 CALL LIST,RETURNS(999)
   90 CALL CHANGE,RETURNS(95)
   95 IF(ICHGCD.EQ.1) GO TO 1
      GO TO 999
  100 IPROC=1
  999 IF(IPROC.EQ.1) RETURN N
      IF(ICHGCD.EQ.1) GO TO 90
      RETURN M
      END
C*****************************    L I S T    ****************************C
C                                                                       C
C     THIS SUBROUTINE LISTS CURRENT PARAMETER VALUES.                   C
C                                                                       C
C***********************************************************************C
      SUBROUTINE LIST,RETURNS(M)
      INTEGER TOPMG,BOTMG,PGNUM,DIVSPC,CHKFNP,OUT
      REAL  MARG(137)
      COMMON /C1000/ CHKFNP, PGNUM, NCOL, LFTMG, MXLINE, TOPMG, BOTMG,
     *               NLPRT, DIVSPC, MNLPG, MARG, NLINE, NSUM1, NSUM2,
     *               MT1, MT2, NPAGE, NCENPG
```

```
      COMMON /C2000/ KR, KW, IN, OUT
      DATA DASH/1H-/, BLANK/1H /, STAR/1H*/
      WRITE(OUT,5)
    5 FORMAT(46H  LIST OF CURRENT PARAMETER VALUES FOR LAYOUT:)
C
      WRITE(OUT,10) CHKFNP
   10 FORMAT(9X,14H1. CHKFNP  =  ,I1)
C
      WRITE(OUT,20) PGNUM
   20 FORMAT(9X,13H2. PGNUM   = ,1X,I1)
C
      WRITE(OUT,30) NCOL
   30 FORMAT(9X,12H3. NCOL    =,I3,8H COLUMNS )
C
      WRITE(OUT,40) LFTMG
   40 FORMAT(9X,12H4. LFTMG   =,I3,8H COLUMNS)
C
      WRITE(OUT,50) MXLINE
   50 FORMAT(9X,13H5. MXLINE  = ,I2,6H LINES)
C
      WRITE(OUT,60) TOPMG
   60 FORMAT(9X,13H6. TOPMG   = ,I2,6H LINES)
C
      WRITE(OUT,70) BOTMG
   70 FORMAT(9X,13H7. BOTMG   = ,I2,6H LINES)
C
      WRITE(OUT,80) NLPRT
   80 FORMAT(9X,13H8. NLPRT   = ,I2,6H LINES)
C
      WRITE(OUT,90) DIVSPC
   90 FORMAT(9X,13H9. DIVSPC  = ,I2,6H LINES)
C
      WRITE(OUT,100) MNLPG
  100 FORMAT(8X,14H10. MNLPG   = ,I2,6H LINES)
C
C
      RETURN M
      END
C*****************************    C H A N G E    *************************C
C                                                                          C
C     THIS SUBROUTINE CHANGES PARAMETER VALUES.                            C
C                                                                          C
C**************************************************************************C
      SUBROUTINE CHANGE,RETURNS(M)
      DIMENSION ANUM(10),INUM(10),PARAM(10),OPARAM(10)
      INTEGER TOPMG,BOTMG,PGNUM,DIVSPC,CHKFNP,OUT
      INTEGER PARAM,OPARAM
      REAL MARG(137),MINUS
      COMMON /C1000/ CHKFNP, PGNUM, NCOL, LFTMG, MXLINE, TOPMG, BOTMG,
     *               NLPRT, DIVSPC, MNLPG, MARG, NLINE, NSUM1, NSUM2,
     *               MT1, MT2, NPAGE, NCENPG
      COMMON /C2000/ KR, KW, IN, OUT
      COMMON /C3000/ CARD(80),ICHGCD,ICOL,IPROC
      EQUIVALENCE(OPARAM,CHKFNP)
      DATA ANUM/ 1H0, 1H1, 1H2, 1H3, 1H4, 1H5, 1H6, 1H7, 1H8, 1H9/
      DATA INUM/  0,   1,   2,   3,   4,   5,   6,   7,   8,   9 /
      DATA BLANK/1H /, COMMA/1H,/, MINUS/1H-/, STAR/1H*/
      ICHGCD=1
  100 DO 101 I=1,10
      PARAM(I)=OPARAM(I)
  101 CONTINUE
      IP=1
      ICOL=0
      NUMBER=-999
  105 WRITE(OUT,106)
  106 FORMAT(30H  ENTER NEW LAYOUT PARAMETERS:)
```

```
      READ(IN,107) (CARD(I),I=1,80)
  107 FORMAT(80A1)
C
  110 ICOL=ICOL+1
      IF(ICOL.GT.80) GO TO 300
C
      IF(CARD(ICOL).EQ.BLANK) GO TO 110
      IF(CARD(ICOL).NE.COMMA.AND.CARD(ICOL).NE.STAR) GO TO 200
      IF(CARD(ICOL).EQ.STAR.AND.CARD(ICOL+1).EQ.COMMA) ICOL=ICOL+1
      IP=IP+1
      IF(IP.GT.10) GO TO 300
      GO TO 110
C
  200 CONTINUE
      DO 210 I=1,10
      NN=I
      IF(CARD(ICOL).EQ.ANUM(I)) GO TO 230
  210 CONTINUE
C
      IF(CARD(ICOL).EQ.MINUS.AND.NUMBER.EQ.-999)
     *    CALL ERROR(5,IP),RETURNS(100)
      IF(NUMBER.EQ.-999) RETURN M
C
C
  220 PARAM(IP)=NUMBER
      NUMBER=-999
      IP=IP+1
      IF(CARD(ICOL).EQ.STAR) IP=IP+1
      IF(IP.GT.10) GO TO 300
      GO TO 110
C
  230 CONTINUE
      IF(NUMBER.NE.-999) NUMBER=NUMBER*10+INUM(NN)
      IF(NUMBER.EQ.-999) NUMBER=INUM(NN)
      ICOL=ICOL+1
      IF(ICOL.GT.80) GO TO 220
      GO TO 200
C
  300 CONTINUE
C     CHECK NEW PARAMETER VALUES
C
C     CHKFNP MUST EQUAL 0,1,2,3
      IF(PARAM(1).LT.0.OR.PARAM(1).GT.3)
     *                    CALL ERROR(5,1), RETURNS(100)
C
C     PGNUM MUST EQUAL 0 OR 1
      IF(PARAM(2).NE.0.AND.PARAM(2).NE.1)
     *                    CALL ERROR(5,2),RETURNS(100)
C
C     NCOL MUST BE 1 TO 137
      IF(PARAM(3).LT.1.OR.PARAM(3).GT.137)
     *                    CALL ERROR(5,3),RETURNS(100)
C
C     LFTMG MUST BE 0 TO 136
      IF(PARAM(4).LT.0.OR.PARAM(4).GT.136)
     *                    CALL ERROR(5,4),RETURNS(100)
C
C     CHECK TOTAL OF LFTMG+NCOL...MUST BE 1 TO 137
      NCOLT=PARAM(3)+PARAM(4)
      IF(NCOLT.LT.1.OR.NCOLT.GT.137)
     *                    CALL ERROR(4,0),RETURNS(100)
C
C     MXLINE MUST BE 1 TO 99
      IF(PARAM(5).LT.1.OR.PARAM(5).GT.99)
     *                    CALL ERROR(5,5),RETURNS(100)
C
```

```
C     CHECK TOTAL OF NLPRT+TOPMG+BOTMG IS EQUAL TO MXLINE
      LINTOT=PARAM(6)+PARAM(7)+PARAM(8)
      IF(LINTOT.NE.PARAM(5))  CALL ERROR(3,0),RETURNS(100)
C
C     TOPMG MUST BE 0 TO MXLINE-1
      IF(PARAM(6).LT.0.OR.PARAM(6).GT.(PARAM(5)-1))
     *                    CALL ERROR(5,6),RETURNS(100)
C
C     BOTMG MUST BE 0 TO MXLINE-1
      IF(PARAM(7).LT.0.OR.PARAM(7).GT.(PARAM(5)-1))
     *                    CALL ERROR(5,7),RETURNS(100)
C
C     NLPRT MUST BE 1 TO MXLINE
      IF(PARAM(8).LT.1.OR.PARAM(8).GT.PARAM(5))
     *                    CALL ERROR(5,8),RETURNS(100)
C
C     TOPMG SHOULD BE GREATER THAN 0 IF PGNUM =1
      IF(PARAM(2).EQ.1.AND.PARAM(6).LT.1) PARAM(8)=PARAM(8)-1
      IF(PARAM(2).EQ.1.AND.PARAM(6).LT.1) PARAM(6)=1
C
C     PARAMETER VALUES ARE OK
  400 CONTINUE
      DO 420 I=1,10
      IF(OPARAM(I).NE.PARAM(I)) OPARAM(I)=PARAM(I)
  420 CONTINUE
      NSUM1=TOPMG+NLPRT
      NSUM2=MXLINE-(MNLPG+DIVSPC+BOTMG)
      MT1=TOPMG/2
      MT2=TOPMG-MT1-1
      I1=LFTMG
      DO 410 I=1,I1
  410 MARG(I)=BLANK
      ICHGCD=0
      RETURN M
      END
C*****************************    E R R O R    *****************************C
C                                                                            C
C     THIS SUBROUTINE PRINTS ERROR MESSAGES.                                 C
C                                                                            C
C*****************************************************************************C
      SUBROUTINE ERROR(N,IP),RETURNS(M)
      INTEGER OUT
      COMMON /C2000/ KR, KW, IN, OUT
      GOTO(1,2,3,4,5) N
    1 WRITE(OUT,100)
  100 FORMAT(57H ***ERROR*** ILLEGAL COMMAND, FOR ASSISTANCE ENTER "HELP
     *")
      RETURN M
C
    2 WRITE(OUT,200)
  200 FORMAT(69H ***ERROR*** ILLEGAL CHARACTER IN PARAMETER LIST FOR "CH
     *ANGE" COMMAND )
      RETURN M
C
    3 WRITE(OUT,300)
  300 FORMAT(48H ***ERROR*** MXLINE MUST EQUAL TOPMG+BOTMG+NLPRT )
      RETURN M
C
    4 WRITE(OUT,400)
  400 FORMAT(57H ***ERROR*** NCOL+LFTMG MUST BE LESS THAN OR EQUAL TO 13
     *7)
      RETURN M
C
    5 WRITE(OUT,500) IP
  500 FORMAT(33H ***ERROR*** VALUE FOR PARAMETER ,I2,16H IS OUT OF RANGE
     *)
```

```
      RETURN M
C
      END
C*****************************    H E L P    *****************************C
C                                                                          C
C     THIS SUBROUTINE PRINTS INSTRUCTIONS TO OPERATE THE PROGRAM.          C
C                                                                          C
C**************************************************************************C
      SUBROUTINE HELP,RETURNS(M)
      INTEGER OUT
      COMMON /C2000/ KR, KW, IN, OUT
      WRITE(OUT,10)
   10 FORMAT(2X,20HCOMMANDS FOR LAYOUT:)
C
      WRITE(OUT,20)
   20 FORMAT(52H  A. HELP    - LISTS "LAYOUT" COMMANDS AND PARAMETERS )
C
      WRITE(OUT,30)
   30 FORMAT(44H  B. LIST    - LISTS CURRENT PARAMETER VALUES)
C
      WRITE(OUT,40)
   40 FORMAT(62H  C. CHANGE - ALLOWS CURRENT PARAMETER VALUE(S) TO BE CH
     *ANGED./12X,52H- NEW VALUES MUST BE ENTERED IN THE FOLLOWING ORDER:
     */17X,50H1. CHKFNP  - NEW PAGE CHECK (0=NO CHECK, 1="1" ON,/
     *17X,32H             2="1" OFF, 3="END") /
     *17X,37H2. PGNUM    - PAGE NUMBER (1=YES,0=NO) /
     *17X,39H3. NCOL     - NUMBER OF COLUMNS PER LINE /
     *17X,37H4. LFTMG    - LEFT MARGIN (IN COLUMNS) /
     *17X,48H5. MXLINE   - MAX. NUMBER OF PRINT LINES PER PAGE /
     *17X,34H6. TOPMG    - TOP MARGIN (IN LINES) /
     *17X,37H7. BOTMG    - BOTTOM MARGIN (IN LINES) /
     *17X,45H8. NLPRT    - NUMBER OF LINES PRINTED PER PAGE /
     *17X,49H9. DIVSPC   - SPACE BETWEEN SUBPROGRAMS (IN LINES) /
     *16X,54H10. MNLPG    - MIN. NO. OF LINES TO START SUBPROGRAM ON/
     *17X,25H              CURRENT PAGE)
C
      WRITE(OUT,50)
   50 FORMAT(12X,57H- TO RETAIN CURRENT PARAMETER VALUE, ENTER A STAR FO
     *R THE/  13X,52H CORRESPONDING PARAMETER IF IT PRECEDES A NEW VALUE
     *./12X,49H  FOR EXAMPLE, TO CHANGE THE VALUE OF NCOL TO 80,/
     *,19X,12HENTER:*,*,80)
C
      WRITE(OUT,60)
   60 FORMAT(12X,57H- COMMANDS "HELP" AND "LIST" MAY BE ADDRESSED FROM W
     *ITHIN /14X,18HTHE "CHANGE" MODE.)
C
      WRITE(OUT,70)
   70 FORMAT(12X,60H- TO EXIT FROM THE COMMAND "CHANGE" WITHOUT ENTERING
     * ANY NEW/13X21H VALUES, ENTER A STAR)
C
      WRITE(OUT,80)
   80 FORMAT(69H  D. PROCEED- IS THE LAST COMMAND TO BE ENTERED; PARAMET
     *ER VALUES ARE,/14X,52HSATISFACTORY, STRUCTURING OF INPUT FILE WILL
     * PROCEED)
C
      RETURN M
      END
C*************************    N E W P A G E    *************************C
C                                                                          C
C     THIS SUBROUTINE POSITIONS PRINTING FACE ON A NEW PAGE AND            C
C     PRINTS PAGE NUMBER.                                                  C
C                                                                          C
C**************************************************************************C
      SUBROUTINE NEWPAGE,RETURNS(M)
      INTEGER TOPMG,BOTMG,PGNUM,DIVSPC,CHKFNP,OUT
      DIMENSION FORMT(2)
```

```
      REAL  MARG(137)
      COMMON /C1000/ CHKFNP, PGNUM, NCOL, LFTMG, MXLINE, TOPMG, BOTMG,
     *               NLPRT, DIVSPC, MNLPG, MARG, NLINE, NSUM1, NSUM2,
     *               MT1, MT2, NPAGE, NCENPG
      COMMON /C2000/ KR, KW, IN, OUT
      DATA FT0/1H(/, FT1/1H /, FT2/1HX/, FT3/6H,1H-,I/, FT4/1H /,
     *     FT5/5H,1H-)/, ONE/1H1/, TWO/1H2/, THREE/1H3/
      IF(NLINE.GE.MXLINE) GO TO 11
      NLINE=NLINE+1
      IF(NLINE.GE.MXLINE) GO TO 11
      DO 10 I=NLINE,MXLINE
      WRITE(KW,5)
    5 FORMAT(1X)
   10 CONTINUE
   11 CONTINUE
      NPAGE=NPAGE+1
      IF(PGNUM.NE.1) GO TO 40
      IF(MT1.LE.0) GO TO 20
      DO 19 I=1,MT1
   19 WRITE(KW,5)
   20 CONTINUE
      NCENPG=NCOL/2+LFTMG-1
      IF(NCENPG.LT.10) ENCODE(1,21,FT1) NCENPG
   21 FORMAT(I1)
      IF(NCENPG.GE.10.AND.NCENPG.LE.99) ENCODE(2,22,FT1) NCENPG
   22 FORMAT(I2)
      IF(NPAGE.LT.10) FT4=ONE
      IF(NPAGE.GE.10.AND.NPAGE.LE.99) FT4=TWO
      IF(NPAGE.GE.100.AND.NPAGE.LE.999) FT4=THREE
      ENCODE(20,23,FORMT) FT0,FT1,FT2,FT3,FT4,FT5
   23 FORMAT(A1,A2,A2,A6,A1,A5)
      WRITE(KW,FORMT) NPAGE
C
      IF(MT2.LE.0) GO TO 31
      DO 30 I=1,MT2
   30 WRITE(KW,5)
   31 CONTINUE
      NLINE=TOPMG
      RETURN M
C
   40 IF(TOPMG.LE.0) GO TO 41
      DO 50 I=1,TOPMG
   50 WRITE(KW,5)
   41 NLINE=TOPMG
      RETURN M
      END
C*************************     E N D C H K     *************************C
C                                                                       C
C     THIS SUBROUTINE CHECKS FOR "END" STATEMENT OF PROGRAM SECTIONS.   C
C                                                                       C
C***********************************************************************C
      SUBROUTINE ENDCHK,RETURNS(N1,N2)
      REAL MARG(137)
      INTEGER TOPMG,BOTMG,PGNUM,DIVSPC,CHKFNP,OUT,ENDFLG
      DIMENSION END(3)
      COMMON /C1000/ CHKFNP, PGNUM, NCOL, LFTMG, MXLINE, TOPMG, BOTMG,
     *               NLPRT, DIVSPC, MNLPG, MARG, NLINE, NSUM1, NSUM2,
     *               MT1, MT2, NPAGE, NCENPG
      COMMON /C2000/ KR, KW, IN, OUT
      COMMON /C4000/ CARD(137),ENDFLG,NBCOL
      DATA BLANK/1H /, END/1HE, 1HN, 1HD/
      ENDFLG=0
      DO 10 I=1,6
      IF(CARD(I).NE.BLANK) RETURN N2
   10 CONTINUE
C
```

```
C     CHECK COL. 7 TO 72 FOR "END",ALL OTHER COLUMNS MUST BE BLANK
      NB=70
      IF(NBCOL.LT.70) NB=NBCOL
      ICOL=6
      DO 20 I=7,NB
      ICOL=ICOL+1
      IF(CARD(ICOL).EQ.BLANK) GO TO 20
      IF(ENDFLG.EQ.1) GO TO 50
C     CHECK FOR "END"
      IF(CARD(ICOL).NE.END(1).OR.
     *   CARD(ICOL+1).NE.END(2).OR.
     *   CARD(ICOL+2).NE.END(3))  GO TO 50
C     "END" LOCATED
      ENDFLG=1
      ICOL=ICOL+2
C     CONTINUE CHECKING FOR VALIDITY OF "END"
C
   20 CONTINUE
      IF(ENDFLG.NE.1) RETURN N2
C
C     VALID "END" STATEMENT LOCATED
      IF(NBCOL.EQ.0) NBCOL=1
      IF(LFTMG.GT.0) WRITE(KW,3) (MARG(I),I=1,LFTMG),(CARD(I),I=1,NBCOL)
      IF(LFTMG.LE.0) WRITE(KW,3) (CARD(I),I=1,NBCOL)
    3 FORMAT(137A1)
      NLINE=NLINE+1
      IF(NLINE.GE.NSUM1.OR.NLINE.GE.NSUM2) CALL NEWPAGE, RETURNS(45)
C
C     INSERT BLANK LINES BETWEEN SUBPROGRAMS
      IF(DIVSPC.LE.0) GO TO 45
      DO 40 I=1,DIVSPC
      IF(NLINE.GT.NSUM1) CALL NEWPAGE,RETURNS(45)
      WRITE(KW,4)
    4 FORMAT(1X)
   40 NLINE=NLINE+1
C
   45 ENDFLG=0
      RETURN N1
C
   50 ENDFLG=0
      RETURN N2
C
      END
 END OF INFORMATION ENCOUNTERED.
/
```

Annexe B
Version non structurée d'un tableau de données

1 DOCUMENTATION AND CALCULATIONS
****

RUNOFF MEASUREMENTS PINE POINT NEAR W17, OCT.19-22,1978

PROPELLOR DESCRIPTIONS FOR CURRENT METER USED:

PROPELLOR 1 (CODE: 19279-6) HAS EQUATIONS
V= 0.0955*N + 0.0600 FOR N < 3.4000
V= 0.1035*N + 0.0300 FOR N >= 3.4000

PROPELLOR 2 (CODE: 584-2) HAS EQUATIONS
V= 0.0983*N + 0.0460 FOR N < 4.4000
V= 0.1007*N + 0.0350 FOR N >= 4.4000

PROPELLOR 3 (CODE: 23295-3) HAS EQUATIONS
V= 0.2298*N + 0.0520 FOR N < 1.6000
V= 0.2570*N + 0.0100 FOR N >= 1.6000

EXPLANATION OF VARIABLES SHOWN IN TABLES:

CM=CENTIMETRES
DEPTH=DEPTH MEASURED FROM BOTTOM OF CREEK TO WATER SURFACE, LAST NUMBER IS TOTAL DEPTH OF CREEK
DISTANCE=DISTANCE MEASURED FROM LEFT TO RIGHT BANK, LOOKING UPSTREAM, FIRST DISTANCE IN CROSSECTION STANDS FOR LEFT BANK, LAST DISTANCE FOR RIGHT BANK
L=LITRES
M=METRES
N=NUMBER OF REVOLUTIONS/SECOND, AS MEASURED
PROP=PROPELLOR NUMBER
REV=REVOLUTIONS
S=SECONDS
V=VELOCITY IN METRES/SECOND

NUMBER OF RUNOFF STATIONS: 3

1 1.DISCHG.CHAN. W17, .5 MI. SOUTH OF RESOLUTION HIGHWAY, OCT.19/78 T:14.45

NUMBER OF MEASUREMENT PROFILES: 11

CROSSECTION OF THE CREEK:

LEFT TO RIGHT BANK IN CM, LOOKING UPSTREAM, TOTAL WIDTH = 265.0 CM

DISTANCE	(CM)	75.0	90.0	110.0	130.0	150.0	165.0	185.0
DEPTH	(CM)	26.0	31.0	50.0	63.0	78.0	106.0	137.0
DISTANCE	(CM)	205.0	230.0	260.0	270.0	300.0	330.0	340.0
DEPTH	(CM)	139.0	124.0	110.0	76.0	68.0	35.0	30.0

MEASUREMENTS AND RESULTS:

DISTANCE CM	DEPTH CM	REV	SECONDS	REV/S	PROP	VELOCITY CM/S
90.0	5.0	20.	25.0	0.80	3	23.6
	15.0	25.	25.0	1.00	3	28.2
	25.0	29.	25.0	1.16	3	31.9
	31.0	--VELOCITY AREA IS 0.08385 M**2/S--				
110.0	5.0	23.	25.0	0.92	3	26.3
	15.0	34.	25.0	1.36	3	36.5
	25.0	36.	25.0	1.44	3	38.3
	35.0	43.	25.0	1.72	3	45.2
	45.0	42.	25.0	1.68	3	44.2
	50.0	--VELOCITY AREA IS 0.18704 M**2/S--				
130.0	5.0	27.	25.0	1.08	3	30.0
	15.0	42.	25.0	1.68	3	44.2
	25.0	47.	25.0	1.88	3	49.3
	35.0	52.	25.0	2.08	3	54.5
	45.0	51	25.0	2.04	3	53.4
	55.0	51.	25.0	2.04	3	53.4
	63.0	--VELOCITY AREA IS 0.29710 M**2/S--				
150.0	5.0	37.	25.0	1.48	3	39.2
	15.0	50.	25.0	2.00	3	52.4
	25.0	56.	25.0	2.24	3	58.6
	40.0	59.	25.0	2.36	3	61.7
	55.0	56.	25.0	2.24	3	58.6
	65.0	53.	25.0	2.12	3	55.5
	75.0	49.	25.0	1.96	3	51.4
	78.0	--VELOCITY AREA IS 0.42200 M**2/S--				
165.0	10.0	30.	25.0	1.20	3	32.8
	30.0	60.	25.0	2.40	3	62.7
	50.0	67.	25.0	2.68	3	69.9
	70.0	62.	25.0	2.48	3	64.7
	90.0	53.	25.0	2.12	3	55.5
	101.0	50.	25.0	2.00	3	52.4
	106.0	--VELOCITY AREA IS 0.59261 M**2/S--				
185.0	10.0	33.	25.0	1.32	3	35.5
	30.0	55.	25.0	2.20	3	57.5
	50.0	67.	25.0	2.68	3	69.9
	70.0	69.	25.0	2.76	3	71.9
	90.0	67.	25.0	2.68	3	69.9
	110.0	58.	25.0	2.32	3	60.6
	130.0	52.	25.0	2.08	3	54.5
	137.0	--VELOCITY AREA IS 0.81370 M**2/S--				
205.0	10.0	37.	25.0	1.48	3	39.2
	30.0	51.	25.0	2.04	3	53.4
	50.0	61	25.0	2.44	3	63.7
	70.0	69.	25.0	2.76	3	71.9
	90.0	68.	25.0	2.72	3	70.9
	110.0	65.	25.0	2.60	3	67.8
	130.0	61.	25.0	2.44	3	63.7
	139.0	--VELOCITY AREA IS 0.84441 M**2/S--				
230.0	10.0	41.	25.0	1.64	3	43.1
	30.0	40.	25.0	1.60	3	42.1
	50.0	54.	25.0	2.16	3	56.5
	70.0	62.	25.0	2.48	3	64.7
	90.0	61.	25.0	2.44	3	63.7
	110.0	61.	25.0	2.44	3	63.7
	120.0	62.	25.0	2.48	3	64.7
	124.0	--VELOCITY AREA IS 0.58349 M**2/S--				
270.0	10.0	31.	25.0	1.24	3	33.7
	30.0	62.	25.0	2.48	3	64.7
	50.0	60.	25.0	2.40	3	62.7
	70.0	57.	25.0	2.28	3	59.6
	76.0	--VELOCITY AREA IS 0.40887 M**2/S--				
300.0	10.0	15.	25.0	0.60	3	19.0

	25.0	22.	25.0	0.88	3	25.4
	40.0	48.	25.0	1.92	3	50.3
	50.0	51.	25.0	2.04	3	53.4
	60.0	49.	25.0	1.96	3	51.4
	68.0	--VELOCITY AREA IS 0.24910 M**2/S--				
330.0	10.0	24.	25.0	0.96	3	27.3
	15.0	37.	25.0	1.48	3	39.2
	30.0	35.	25.0	1.40	3	37.4
	35.0	--VELOCITY AREA IS 0.10963 M**2/S--				

THE RUNOFF IS 1103.7 L/S .

1

2.DISCHARGE CHANNEL W17, AT RESOLUTION HIGHWAY OCT.19/78 T:17.00

NUMBER OF MEASUREMENT PROFILES: 13

CROSSECTION OF THE CREEK:

LEFT TO RIGHT BANK IN CM, LOOKING UPSTREAM, TOTAL WIDTH = 420.0 CM

DISTANCE (CM)	20.0	40.0	70.0	100.0	130.0	160.0	190.0
DEPTH (CM)	0.0	21.0	38.0	53.0	71.0	85.0	90.0
DISTANCE (CM)	220.0	250.0	280.0	310.0	340.0	370.0	400.0
DEPTH (CM)	93.0	92.0	86.0	80.0	66.0	46.0	32.0
DISTANCE (CM)	440 0						
DEPTH (CM)	0.0						

MEASUREMENTS AND RESULTS:

DISTANCE CM	DEPTH CM	REV	SECONDS	REV/S	PROP	VELOCITY CM/S
40.0	5.0	21.	25.0	0.84	3	24.5
	11.0	32.	25.0	1.28	3	34.6
	17.0	27.	25.0	1.08	3	30.0
	21.0	--VELOCITY AREA IS 0.05771 M**2/S--				
70.0	5.0	18.	25.0	0.72	3	21.7
	15.0	31.	25.0	1.24	3	33.7
	25.0	34.	25.0	1.36	3	36.5
	35.0	38.	25.0	1.52	3	40.1
	38.0	--VELOCITY AREA IS 0.12128 M**2/S--				
100.0	5.0	19.	25.0	0.76	3	22.7
	15.0	28.	25.0	1.12	3	30.9
	25.0	34.	25.0	1.36	3	36.5
	35.0	38.	25.0	1.52	3	40.1
	45.0	41.	25.0	1.64	3	43.1
	53.0	--VELOCITY AREA IS 0.18344 M**2/S--				
130.0	5.0	26.	25.0	1.04	3	29.1
	15.0	39.	25.0	1.56	3	41.0
	25.0	36.	25.0	1.44	3	38.3
	40.0	45.	25.0	1.80	3	47.3
	55.0	48.	25.0	1.92	3	50.3
	67.0	48.	25.0	1.92	3	50.3
	71.0	--VELOCITY AREA IS 0.30357 M**2/S--				
160.0	5.0	29.	25.0	1.16	3	31.9
	15.0	39.	25.0	1.56	3	41 0
	25.0	46.	25.0	1.84	3	48.3

```
          40.0   47.    25.0      1.88      3     49.3
          55.0   55.    25.0      2.20      3     57.5
          70.0   54.    25.0      2.16      3     56.5
          80.0   55.    25.0      2.20      3     57.5
          85.0   --VELOCITY AREA IS 0.41775 M**2/S--
190.0      5.0   28.    25.0      1.12      3     30.9
          15.0   42.    25.0      1.68      3     44.2
          25.0   44.    25.0      1.76      3     46.2
          40.0   54.    25.0      2.16      3     56.5
          55.0   51.    25.0      2.04      3     53.4
          70.0   62.    25.0      2.48      3     64.7
          85.0   59.    25 0      2.36      3     61.7
          90.0   --VELOCITY AREA IS 0.46786 M**2/S--
220.0      5.0   34.    25.0      1.36      3     36.5
          15.0   44.    25.0      1.76      3     46.2
          25.0   43.    25.0      1.72      3     45.2
          40.0   48.    25.0      1.92      3     50.3
          55.0   52.    25.0      2.08      3     54.5
          70.0   59.    25.0      2.36      3     61.7
          85.0   60.    25.0      2.40      3     62.7
          93.0   --VELOCITY AREA IS 0.48147 M**2/S--
250.0      5.0   30.    25.0      1.20      3     32.8
          15.0   35.    25.0      1.40      3     37.4
          25.0   38.    25.0      1.52      3     40.1
          40.0   47.    25.0      1.88      3     49.3
          55.0   50.    25.0      2.00      3     52.4
          70.0   51.    25.0      2.04      3     53.4
          85.0   59.    25.0      2.36      3     61.7
          92.0   --VELOCITY AREA IS 0.43832 M**2/S--
280.0      5.0   39.    25.0      1.56      3     41.0
          15.0   36.    25.0      1.44      3     38.3
          25.0   42     25.0      1.68      3     44.2
          40.0   45.    25.0      1.80      3     47.3
          55.0   52.    25.0      2.08      3     54.5
          70.0   53.    25.0      2.12      3     55.5
          81.0   57.    25.0      2.28      3     59.6
          86.0   --VELOCITY AREA IS 0.41671 M**2/S--
310.0      5.0   25.    25.0      1.00      3     28.2
          15.0   32.    25.0      1.28      3     34.6
          25.0   39     25.0      1.56      3     41.0
          40.0   46.    25.0      1.84      3     48.3
          55.0   43.    25.0      1.72      3     45.2
          65.0   48.    25.0      1.92      3     50.3
          75.0   53.    25.0      2.12      3     55.5
          80.0   --VELOCITY AREA IS 0.34535 M**2/S--
340.0      5.0   11.    25.0      0.44      3     15.3
          15.0   31.    25.0      1.24      3     33.7
          25.0   36.    25.0      1.44      3     38.3
          40.0   41.    25.0      1.64      3     43.1
          51.0   43.    25.0      1.72      3     45.2
          62.0   47.    25.0      1.88      3     49.3
          66.0   --VELOCITY AREA IS 0.24724 M**2/S--
370 0      5.0   23.    25.0      0.92      3     26.3
          15.0   36.    25.0      1.44      3     38.3
          25.0   42.    25.0      1.68      3     44.2
          33.0   43.    25.0      1.72      3     45.2
          42.0   43.    25.0      1.72      3     45.2
          46.0   --VELOCITY AREA IS 0.17795 M**2/S--
400.0      5.0   22.    25.0      0.88      3     25.4
          15.0   28.    25.0      1.12      3     30.9
          27.0   30.    25.0      1.20      3     32.8
          32.0   --VELOCITY AREA IS 0.09233 M**2/S--


          -------------------------
          THE RUNOFF IS 1127.0 L/S .
```

1

3.DISCHG.CHAN. W17, UPSTREAM OF CONFLUENCE XJ5 CHANNEL, OCT.20/78 T:13.15

NUMBER OF MEASUREMENT PROFILES: 13

CROSSECTION OF THE CREEK:

LEFT TO RIGHT BANK IN CM, LOOKING UPSTREAM, TOTAL WIDTH = 440.0 CM

DISTANCE	(CM)	90.0	110.0	140.0	170.0	200.0	230.0	260.0
DEPTH	(CM)	0.0	20.0	32.0	45.0	53.0	65.0	67.0
DISTANCE	(CM)	290.0	320.0	350.0	380.0	410.0	460.0	490.0
DEPTH	(CM)	69.0	72.0	68.0	55.0	57.0	33.0	25.0
DISTANCE	(CM)	530.0						
DEPTH	(CM)	0.0						

MEASUREMENTS AND RESULTS:

DISTANCE CM	DEPTH CM	REV	SECONDS	REV/S	PROP	VELOCITY CM/S
110.0	5.0	25.	25.0	1.00	3	28.2
	10.0	31.	25.0	1.24	3	33.7
	15.0	35.	25.0	1.40	3	37.4
	20.0	--VELOCITY AREA IS 0.06249 M**2/S--				
140.0	5.0	14.	25.0	0.56	3	18.1
	15.0	32.	25.0	1.28	3	34.6
	25.0	39.	25.0	1.56	3	41.0
	32.0	--VELOCITY AREA IS 0.09968 M**2/S--				
170.0	5.0	21.	25.0	0.84	3	24.5
	15.0	36.	25.0	1.44	3	38.3
	25.0	40.	25.0	1.60	3	42.1
	35.0	43.	25.0	1.72	3	45.2
	41.0	45.	25.0	1.80	3	47.3
	45.0	--VELOCITY AREA IS 0.17110 M**2/S--				
200.0	5.0	28.	25.0	1.12	3	30.9
	15.0	42.	25.0	1.68	3	44.2
	25.0	51.	25.0	2.04	3	53.4
	35.0	50.	25.0	2.00	3	52.4
	45.0	48.	25.0	1.92	3	50.3
	53.0	--VELOCITY AREA IS 0.24186 M**2/S--				
230.0	5.0	22.	25.0	0.88	3	25.4
	15.0	41.	25.0	1.64	3	43.1
	25.0	51.	25.0	2.04	3	53.4
	35.0	57.	25.0	2.28	3	59.6
	45.0	56.	25.0	2.24	3	58.6
	60.0	51.	25.0	2.04	3	53.4
	65.0	--VELOCITY AREA IS 0.31798 M**2/S--				
260.0	5.0	17.	25.0	0.68	3	20.8
	15.0	54.	25.0	2.16	3	56.5
	25.0	58.	25.0	2.32	3	60.6
	40.0	60.	25.0	2.40	3	62.7
	50.0	57.	25.0	2.28	3	59.6
	61.0	53.	25.0	2.12	3	55.5
	67.0	--VELOCITY AREA IS 0.35327 M**2/S--				
290.0	5.0	40.	25.0	1.60	3	42.1
	15.0	59.	25.0	2.36	3	61.7
	25.0	63.	25.0	2.52	3	65.8
	40.0	65.	25.0	2.60	3	67.8

```
                  55.0    63.     25.0        2.52      3     65.8
                  65.0    59.     25.0        2.36      3     61.7
                  69.0    --VELOCITY AREA IS 0.41980 M**2/S--
         320.0     5.0    31.     25.0        1.24      3     33.7
                  15.0    55.     25.0        2.20      3     57.5
                  25.0    62.     25.0        2.48      3     64.7
                  40.0    69.     25.0        2.76      3     71.9
                  55.0    69.     25.0        2.76      3     71.9
                  68.0    71.     25.0        2.84      3     74.0
                  72.0    --VELOCITY AREA IS 0.45423 M**2/S--
         350.0     5.0    24.     25.0        0.96      3     27.3
                  15.0    50.     25.0        2.00      3     52.4
                  25.0    59.     25.0        2.36      3     61.7
                  40.0    70.     25.0        2.80      3     73.0
                  52.0    70.     25.0        2.80      3     73.0
                  64.0    71.     25.0        2.84      3     74.0
                  68.0    --VELOCITY AREA IS 0.41335 M**2/S--
         380.0     5.0    37.     25.0        1.48      3     39.2
                  15.0    52.     25.0        2.08      3     54.5
                  25.0    62.     25.0        2.48      3     64.7
                  35.0    69.     25.0        2.76      3     71.9
                  45.0    75.     25.0        3.00      3     78.1
                  51.0    74.     25.0        2.96      3     77.1
                  55.0    --VELOCITY AREA IS 0.34173 M**2/S--
         410.0     5.0    24.     25.0        0.96      3     27.3
                  15.0    38.     25.0        1.52      3     40.1
                  25.0    55.     25.0        2.20      3     57.5
                  35.0    66.     25.0        2.64      3     68.8
                  45.0    67.     25.0        2.68      3     69.9
                  53.0    74.     25.0        2.96      3     77.1
                  57.0    --VELOCITY AREA IS 0.31492 M**2/S--
         460.0     5.0    27.     25.0        1.08      3     30.0
                  15.0    45.     25.0        1.84      3     48.3
                  23.0    46.     25.0        1.84      3     48.3
                  30.0    44.     25.0        1.76      3     46.2
                  33.0    --VELOCITY AREA IS 0.13586 M**2/S--
         490.0     5.0    27.     25.0        1.08      3     30.0
                  13.0    38.     25.0        1.52      3     40.1
                  20.0    36.     25.0        1.44      3     38.3
                  25.0    --VELOCITY AREA IS 0.08558 M**2/S--


                          ---------------------------
                          THE RUNOFF IS 1069.8 L/S .
                          ---------------------------
1                          SUMMARY OF RESULTS
                          ********************

      RUNOFF MEASUREMENTS PINE POINT NEAR W17, OCT.19-22,1978
      ------------------------------------------------------------


  1.DISCHG.CHAN. W17, .5 MI. SOUTH OF RESOLUTION HIGHWAY, OCT.19/78 T:14.45
                  *** THE RUNOFF IS 1103.7 L/S ***

  2.DISCHARGE CHANNEL W17, AT RESOLUTION HIGHWAY   OCT.19/78 T:17.00
                  *** THE RUNOFF IS 1127.0 L/S ***

  3.DISCHG.CHAN. W17, UPSTREAM OF CONFLUENCE X15 CHANNEL, OCT.20/78 T:13.15
                  *** THE RUNOFF IS 1069.8 L/S ***

 END OF INFORMATION ENCOUNTERED.
/
```

Annexe C
Version structurée du programme LAYOUT

```
C*****************************   L A Y O U T   ***************************C
C                                                                           C
C     MAINLINE PROGRAM TO DESIGN PRINTING FACES FOR DOCUMENTATION           C
C     OF COMPUTER FILES PRINTED ON-LINE AT REMOTE TERMINALS.                C
C                                                                           C
C                           AUTHOR - W.C.HORWOOD                            C
C                                                                           C
C*****************************************************************************C
      PROGRAM LAYOUT(INPUT,OUTPUT,A,B,TAPE1=A,TAPE2=B,TAPE5=INPUT,
     *               TAPE6=OUTPUT)
      IMPLICIT INTEGER(D)
      INTEGER TOPMG,BOTMG,PGNUM,DIVSPC,CHKFNP,OUT,ENDFLG
      DIMENSION END(3)
      REAL  MARG(137)
      COMMON /C1000/ CHKFNP, PGNUM, NCOL, LFTMG, MXLINE, TOPMG, BOTMG,
     *              NLPRT, DIVSPC, MNLPG, MARG, NLINE, NSUM1, NSUM2,
     *              MT1, MT2, NPAGE, NCENPG
      COMMON /C2000/ KR, KW, IN, OUT
      COMMON /C3000/ CCARD(80),ICHGCD,ICOL,IPROC
      COMMON /C4000/ CARD(137), ENDFLG, NBCOL
      DATA CC1/1H1/, BLANK/1H /
      DATA END/1HE, 1HN, 1HD/,  C/1HC/
      DATA DTOPMG/6/,   DNLPRT/55/,   DBOTMG/5/,   DMXLINE/66/,
     *     DDIVSPC/6/, DMNLPG/9/,    DPGNUM/1/,   DNCOL /72/,
     *     DLFTMG/10/, DCHKFNP/3/
      DATA KR/1/, KW/2/, IN/5/, OUT/6/
C
      NPAGE=0
      ICHGCD=0
C
C.....INITIALIZE VALUES
      TOPMG=DTOPMG
      BOTMG=DBOTMG
      NCOL =DNCOL
      NLPRT=DNLPRT
      MXLINE=DMXLINE
      DIVSPC=DDIVSPC
      MNLPG=DMNLPG
      PGNUM=DPGNUM
      LFTMG=DLFTMG
      DO 5 I=1,LFTMG
    5 MARG(I)=BLANK
      CHKFNP=DCHKFNP
      NSUM1=TOPMG+NLPRT
      NSUM2=MXLINE-(MNLPG+DIVSPC+BOTMG)
      MT1=TOPMG/2
      MT2=TOPMG-MT1-1
C
      WRITE(OUT,6)
    6 FORMAT(/27(1H*),17H   L A Y O U T   ,28(1H*)/
     * 24H  ENTER LAYOUT COMMANDS:/2X,21(1H-))
   99 CONTINUE
      READ(IN,7) CCARD
    7 FORMAT(80A1)
      CALL RDCMND,RETURNS(99,8)
```

```
    8 CONTINUE
      WRITE(OUT,9)
    9 FORMAT(/14(1H*),44H   PROGRAM PROCEEDS WITH FILE STRUCTURING
     * ,14(1H*)/)
C
C
C.....POSITION PAPER TO PAGE ONE, FIRST PRINTED LINE
      NPAGE=NPAGE+1
      NCENPG=NCOL/2+LFTMG
      IF(TOPMG.LE.0) GO TO 11
      DO 10 I=1,TOPMG
   10 WRITE(KW,1)
    1 FORMAT(1X)
   11 CONTINUE
      NLINE=TOPMG
C
C.....READ NEXT CARD
  100 READ(KR,2) (CARD(I),I=1,NCOL )
    2 FORMAT(137A1)
      IF(EOF(KR)) 999,110
C
  110 CONTINUE
      DTBLNK=NCOL+1
      DO 115 I=1,NCOL
      DTBLNK=DTBLNK-1
      IF(CARD(DTBLNK).NE.BLANK) GO TO 116
  115 CONTINUE
  116 CONTINUE
      NBCOL=DTBLNK
C
C     CHECK FOR "C END" IN COLS. 1 TO 6
      IF(CARD(1).NE.C) GO TO 119
      I=1
  117 I=I+1
      IF(I.GT.4) GO TO 119
      IF(CARD(I).EQ.BLANK) GO TO 117
      IF(CARD(I).NE.END(1).OR.CARD(I+1).NE.END(2).OR.
     *   CARD(I+2).NE.END(3)) GO TO 119
      I1=I+3
      DO 118 I=I1,NBCOL
      IF(CARD(I).NE.BLANK) GO TO 119
  118 CONTINUE
      CALL NEWPAGE,RETURNS(100)
C
  119 CONTINUE
C.....CHECK OPTIONS FOR CHKFNP
      IF(CHKFNP.EQ.0) GO TO 200
      IF(CHKFNP.EQ.3) CALL ENDCHK,RETURNS(100,200)
C     CHKFNP=1 OR 2...CHECK FOR CARRIAGE CONTROL CHARACTERS
      IF(CARD(1).NE.CC1) GO TO 200
C
      DO 130 I=2,NBCOL
  130 CARD(I-1)=CARD(I)
      NBCOL=NBCOL-1
      IF(NBCOL.EQ.0) NBCOL=1
```

```
C
      IF(CHKFNP.EQ.1.AND.NLINE.GT.TOPMG) CALL NEWPAGE,RETURNS(200)
C
  200 CONTINUE
      IF(NLINE.GE.NSUM1) CALL NEWPAGE,RETURNS(205)
      IF(NBCOL.EQ.0) GO TO 100
  205 CONTINUE
      IF(LFTMG.GT.0) WRITE(KW,3) (MARG(I),I=1,LFTMG),(CARD(I),I=1,NBCOL)
      IF(LFTMG.LE.0) WRITE(KW,3) (CARD(I),I=1,NBCOL)
      NLINE=NLINE+1
      GO TO 100
    3 FORMAT(137A1)
C
C     SKIP TO NEWPAGE AT END
  999 CONTINUE
      IF(NLINE.GE.(TOPMG+1)) CALL NEWPAGE,RETURNS(1000)
 1000 STOP
      END
```

```
C*************************    R D C M N D    ************************C
C                                                                     C
C     THIS SUBROUTINE READS COMMANDS AND CALLS APPROPRIATE            C
C     SUBROUTINES.                                                    C
C                                                                     C
C*********************************************************************C
      SUBROUTINE RDCMND,RETURNS(M,N)
      DIMENSION CMND(4,4)
      INTEGER TOPMG,BOTMG,PGNUM,DIVSPC,CHKFNP,OUT
      REAL MARG(137)
      COMMON /C1000/ CHKFNP, PGNUM, NCOL, LFTMG, MXLINE, TOPMG, BOTMG,
     *               NLPRT, DIVSPC, MNLPG, MARG, NLINE, NSUM1, NSUM2,
     *               MT1, MT2, NPAGE, NCENPG
      COMMON /C2000/ KR,KW,IN,OUT
      COMMON /C3000/ CCARD(80),ICHGCD,ICOL,IPROC
      DATA CMND/ 1HH, 1HL, 1HC, 1HP,
     *           1HE, 1HI, 1HH, 1HR,
     *           1HL, 1HS, 1HA, 1HO,
     *           1HP, 1HT, 1HN, 1HC /
      DATA BLANK/1H /
C
    1 CONTINUE
      IPROC=0
C     SKIP BLANKS
      ICOL=1
      DO 10 I=1,76
      ICOL=I
      IF(CCARD(I).NE.BLANK) GO TO 20
   10 CONTINUE
      CALL ERROR(1,0),RETURNS(999)
C
C     READ COMMAND
   20 CONTINUE
      DO 30 J=1,4
      ICMND=J
      IF(CCARD(ICOL).EQ.CMND(J,1)) GO TO 40
   30 CONTINUE
      IF(ICHGCD.EQ.1) CALL ERROR(2,0),RETURNS(999)
      CALL ERROR(1,0),RETURNS(999)
C
   40 DO 50 K=2,4
      ICOL=ICOL+1
      IF(CCARD(ICOL).EQ.CMND(ICMND,K)) GO TO 50
C     INVALID COMMAND
      CALL ERROR(1,0),RETURNS(999)
   50 CONTINUE
C
C     VALID COMMAND LOCATED
      GOTO(70,80,90,100),ICMND
   70 CALL HELP,RETURNS(999)
   80 CALL LIST,RETURNS(999)
   90 CALL CHANGE,RETURNS(95)
   95 IF(ICHGCD.EQ.1) GO TO 1
      GO TO 999
  100 IPROC=1
```

```
999 IF(IPROC.EQ.1) RETURN N
    IF(ICHGCD.EQ.1) GO TO 90
    RETURN M
    END
```

```
C*****************************    L I S T    ***************************C
C                                                                       C
C     THIS SUBROUTINE LISTS CURRENT PARAMETER VALUES.                   C
C                                                                       C
C***********************************************************************C
      SUBROUTINE LIST,RETURNS(M)
      INTEGER TOPMG,BOTMG,PGNUM,DIVSPC,CHKFNP,OUT
      REAL  MARG(137)
      COMMON /C1000/ CHKFNP, PGNUM, NCOL, LFTMG, MXLINE, TOPMG, BOTMG,
     *               NLPRT, DIVSPC, MNLPG, MARG, NLINE, NSUM1, NSUM2,
     *               MT1, MT2, NPAGE, NCENPG
      COMMON /C2000/ KR, KW, IN, OUT
      DATA DASH/1H-/, BLANK/1H /, STAR/1H*/
      WRITE(OUT,5)
    5 FORMAT(46H  LIST OF CURRENT PARAMETER VALUES FOR LAYOUT:)
C
      WRITE(OUT,10) CHKFNP
   10 FORMAT(9X,14H1. CHKFNP  =  ,I1)
C
      WRITE(OUT,20) PGNUM
   20 FORMAT(9X,13H2. PGNUM   = ,1X,I1)
C
      WRITE(OUT,30) NCOL
   30 FORMAT(9X,12H3. NCOL    =,I3,8H COLUMNS )
C
      WRITE(OUT,40) LFTMG
   40 FORMAT(9X,12H4. LFTMG   =,I3,8H COLUMNS)
C
      WRITE(OUT,50) MXLINE
   50 FORMAT(9X,13H5. MXLINE  = ,I2,6H LINES)
C
      WRITE(OUT,60) TOPMG
   60 FORMAT(9X,13H6. TOPMG   = ,I2,6H LINES)
C
      WRITE(OUT,70) BOTMG
   70 FORMAT(9X,13H7. BOTMG   = ,I2,6H LINES)
C
      WRITE(OUT,80) NLPRT
   80 FORMAT(9X,13H8. NLPRT   = ,I2,6H LINES)
C
      WRITE(OUT,90) DIVSPC
   90 FORMAT(9X,13H9. DIVSPC  = ,I2,6H LINES)
C
      WRITE(OUT,100) MNLPG
  100 FORMAT(8X,14H10. MNLPG   = ,I2,6H LINES)
C
C
      RETURN M
      END
```

```
C*************************************    C H A N G E    ************************C
C                                                                               C
C     THIS SUBROUTINE CHANGES PARAMETER VALUES.                                 C
C                                                                               C
C*******************************************************************************C
      SUBROUTINE CHANGE,RETURNS(M)
      DIMENSION ANUM(10),INUM(10),PARAM(10),OPARAM(10)
      INTEGER TOPMG,BOTMG,PGNUM,DIVSPC,CHKFNP,OUT
      INTEGER PARAM,OPARAM
      REAL MARG(137),MINUS
      COMMON /C1000/ CHKFNP, PGNUM, NCOL, LFTMG, MXLINE, TOPMG, BOTMG,
     *               NLPRT, DIVSPC, MNLPG, MARG, NLINE, NSUM1, NSUM2,
     *               MT1, MT2, NPAGE, NCENPG
      COMMON /C2000/ KR, KW, IN, OUT
      COMMON /C3000/ CARD(80),ICHGCD,ICOL,IPROC
      EQUIVALENCE(OPARAM,CHKFNP)
      DATA ANUM/ 1H0, 1H1, 1H2, 1H3, 1H4, 1H5, 1H6, 1H7, 1H8, 1H9/
      DATA INUM/  0,   1,   2,   3,   4,   5,   6,   7,   8,   9 /
      DATA BLANK/1H /, COMMA/1H,/, MINUS/1H-/, STAR/1H*/
      ICHGCD=1
  100 DO 101 I=1,10
      PARAM(I)=OPARAM(I)
  101 CONTINUE
      IP=1
      ICOL=0
      NUMBER=-999
  105 WRITE(OUT,106)
  106 FORMAT(30H  ENTER NEW LAYOUT PARAMETERS:)
      READ(IN,107) (CARD(I),I=1,80)
  107 FORMAT(80A1)
C
  110 ICOL=ICOL+1
      IF(ICOL.GT.80) GO TO 300
C
      IF(CARD(ICOL).EQ.BLANK) GO TO 110
      IF(CARD(ICOL).NE.COMMA.AND.CARD(ICOL).NE.STAR) GO TO 200
      IF(CARD(ICOL).EQ.STAR.AND.CARD(ICOL+1).EQ.COMMA) ICOL=ICOL+1
      IP=IP+1
      IF(IP.GT.10) GO TO 300
      GO TO 110
C
  200 CONTINUE
      DO 210 I=1,10
      NN=I
      IF(CARD(ICOL).EQ.ANUM(I)) GO TO 230
  210 CONTINUE
C
      IF(CARD(ICOL).EQ.MINUS.AND.NUMBER.EQ.-999)
     *    CALL ERROR(5,IP),RETURNS(100)
      IF(NUMBER.EQ.-999) RETURN M
C
C
  220 PARAM(IP)=NUMBER
      NUMBER=-999
      IP=IP+1
```

```
      IF(CARD(ICOL).EQ.STAR) IP=IP+1
      IF(IP.GT.10) GO TO 300
      GO TO 110
C
  230 CONTINUE
      IF(NUMBER.NE.-999) NUMBER=NUMBER*10+INUM(NN)
      IF(NUMBER.EQ.-999) NUMBER=INUM(NN)
      ICOL=ICOL+1
      IF(ICOL.GT.80) GO TO 220
      GO TO 200
C
  300 CONTINUE
C     CHECK NEW PARAMETER VALUES
C
C     CHKFNP MUST EQUAL 0,1,2,3
      IF(PARAM(1).LT.0.OR.PARAM(1).GT.3)
     *                    CALL ERROR(5,1), RETURNS(100)
C
C     PGNUM MUST EQUAL 0 OR 1
      IF(PARAM(2).NE.0.AND.PARAM(2).NE.1)
     *                    CALL ERROR(5,2),RETURNS(100)
C
C     NCOL MUST BE 1 TO 137
      IF(PARAM(3).LT.1.OR.PARAM(3).GT.137)
     *                    CALL ERROR(5,3),RETURNS(100)
C
C     LFTMG MUST BE 0 TO 136
      IF(PARAM(4).LT.0.OR.PARAM(4).GT.136)
     *                    CALL ERROR(5,4),RETURNS(100)
C
C     CHECK TOTAL OF LFTMG+NCOL...MUST BE 1 TO 137
      NCOLT=PARAM(3)+PARAM(4)
      IF(NCOLT.LT.1.OR.NCOLT.GT.137)
     *                    CALL ERROR(4,0),RETURNS(100)
C
C     MXLINE MUST BE 1 TO 99
      IF(PARAM(5).LT.1.OR.PARAM(5).GT.99)
     *                    CALL ERROR(5,5),RETURNS(100)
C
C     CHECK TOTAL OF NLPRT+TOPMG+BOTMG IS EQUAL TO MXLINE
      LINTOT=PARAM(6)+PARAM(7)+PARAM(8)
      IF(LINTOT.NE.PARAM(5))  CALL ERROR(3,0),RETURNS(100)
C
C     TOPMG MUST BE 0 TO MXLINE-1
      IF(PARAM(6).LT.0.OR.PARAM(6).GT.(PARAM(5)-1))
     *                    CALL ERROR(5,6),RETURNS(100)
C
C     BOTMG MUST BE 0 TO MXLINE-1
      IF(PARAM(7).LT.0.OR.PARAM(7).GT.(PARAM(5)-1))
     *                    CALL ERROR(5,7),RETURNS(100)
C
C     NLPRT MUST BE 1 TO MXLINE
      IF(PARAM(8).LT.1.OR.PARAM(8).GT.PARAM(5))
     *                    CALL ERROR(5,8),RETURNS(100)
C
```

```
C     TOPMG SHOULD BE GREATER THAN 0 IF PGNUM =1
      IF(PARAM(2).EQ.1.AND.PARAM(6).LT.1) PARAM(8)=PARAM(8)-1
      IF(PARAM(2).EQ.1.AND.PARAM(6).LT.1) PARAM(6)=1
C
C     PARAMETER VALUES ARE OK
  400 CONTINUE
      DO 420 I=1,10
      IF(OPARAM(I).NE.PARAM(I)) OPARAM(I)=PARAM(I)
  420 CONTINUE
      NSUM1=TOPMG+NLPRT
      NSUM2=MXLINE-(MNLPG+DIVSPC+BOTMG)
      MT1=TOPMG/2
      MT2=TOPMG-MT1-1
      I1=LFTMG
      DO 410 I=1,I1
  410 MARG(I)=BLANK
      ICHGCD=0
      RETURN M
      END
```

```
C*****************************    E R R O R    ****************************C
C                                                                           C
C     THIS SUBROUTINE PRINTS ERROR MESSAGES.                                C
C                                                                           C
C***************************************************************************C
      SUBROUTINE ERROR(N,IP),RETURNS(M)
      INTEGER OUT
      COMMON /C2000/ KR, KW, IN, OUT
      GOTO(1,2,3,4,5) N
    1 WRITE(OUT,100)
  100 FORMAT(57H ***ERROR*** ILLEGAL COMMAND, FOR ASSISTANCE ENTER "HELP
     *")
      RETURN M
C
    2 WRITE(OUT,200)
  200 FORMAT(69H ***ERROR*** ILLEGAL CHARACTER IN PARAMETER LIST FOR "CH
     *ANGE" COMMAND )
      RETURN M
C
    3 WRITE(OUT,300)
  300 FORMAT(48H ***ERROR*** MXLINE MUST EQUAL TOPMG+BOTMG+NLPRT )
      RETURN M
C
    4 WRITE(OUT,400)
  400 FORMAT(57H ***ERROR*** NCOL+LFTMG MUST BE LESS THAN OR EQUAL TO 13
     *7)
      RETURN M
C
    5 WRITE(OUT,500) IP
  500 FORMAT(33H ***ERROR*** VALUE FOR PARAMETER ,I2,16H IS OUT OF RANGE
     *)
      RETURN M
C
      END
```

```
C*****************************    H E L P    ****************************C
C                                                                         C
C     THIS SUBROUTINE PRINTS INSTRUCTIONS TO OPERATE THE PROGRAM.         C
C                                                                         C
C*************************************************************************C
      SUBROUTINE HELP,RETURNS(M)
      INTEGER OUT
      COMMON /C2000/ KR, KW, IN, OUT
      WRITE(OUT,10)
   10 FORMAT(2X,20HCOMMANDS FOR LAYOUT:)
C
      WRITE(OUT,20)
   20 FORMAT(52H  A. HELP   - LISTS "LAYOUT" COMMANDS AND PARAMETERS )
C
      WRITE(OUT,30)
   30 FORMAT(44H  B. LIST   - LISTS CURRENT PARAMETER VALUES)
C
      WRITE(OUT,40)
   40 FORMAT(62H  C. CHANGE - ALLOWS CURRENT PARAMETER VALUE(S) TO BE CH
     *ANGED./12X,52H- NEW VALUES MUST BE ENTERED IN THE FOLLOWING ORDER:
     */17X,50H1. CHKFNP  - NEW PAGE CHECK (0=NO CHECK, 1="1" ON,/
     *17X,32H              2="1" OFF, 3="END") /
     *17X,37H2. PGNUM    - PAGE NUMBER (1=YES,0=NO) /
     *17X,39H3. NCOL     - NUMBER OF COLUMNS PER LINE /
     *17X,37H4. LFTMG    - LEFT MARGIN (IN COLUMNS) /
     *17X,48H5. MXLINE   - MAX. NUMBER OF PRINT LINES PER PAGE /
     *17X,34H6. TOPMG    - TOP MARGIN (IN LINES) /
     *17X,37H7. BOTMG    - BOTTOM MARGIN (IN LINES) /
     *17X,45H8. NLPRT    - NUMBER OF LINES PRINTED PER PAGE /
     *17X,49H9. DIVSPC   - SPACE BETWEEN SUBPROGRAMS (IN LINES) /
     *16X,54H10. MNLPG    - MIN. NO. OF LINES TO START SUBPROGRAM ON/
     *17X,25H              CURRENT PAGE)
C
      WRITE(OUT,50)
   50 FORMAT(12X,57H- TO RETAIN CURRENT PARAMETER VALUE, ENTER A STAR FO
     *R THE/  13X,52H CORRESPONDING PARAMETER IF IT PRECEDES A NEW VALUE
     *./12X,49H  FOR EXAMPLE, TO CHANGE THE VALUE OF NCOL TO 80,/
     *,19X,12HENTER:*,*,80)
C
      WRITE(OUT,60)
   60 FORMAT(12X,57H- COMMANDS "HELP" AND "LIST" MAY BE ADDRESSED FROM W
     *ITHIN /14X,18HTHE "CHANGE" MODE.)
C
      WRITE(OUT,70)
   70 FORMAT(12X,60H- TO EXIT FROM THE COMMAND "CHANGE" WITHOUT ENTERING
     * ANY NEW/13X21H VALUES, ENTER A STAR)
C
      WRITE(OUT,80)
   80 FORMAT(69H  D. PROCEED- IS THE LAST COMMAND TO BE ENTERED; PARAMET
     *ER VALUES ARE,/14X,52HSATISFACTORY, STRUCTURING OF INPUT FILE WILL
     * PROCEED)
C
      RETURN M
      END
```

```
C*************************    N E W P A G E    *************************C
C                                                                       C
C     THIS SUBROUTINE POSITIONS PRINTING FACE ON A NEW PAGE AND         C
C     PRINTS PAGE NUMBER.                                               C
C                                                                       C
C***********************************************************************C
      SUBROUTINE NEWPAGE,RETURNS(M)
      INTEGER TOPMG,BOTMG,PGNUM,DIVSPC,CHKFNP,OUT
      DIMENSION FORMT(2)
      REAL  MARG(137)
      COMMON /C1000/ CHKFNP, PGNUM, NCOL, LFTMG, MXLINE, TOPMG, BOTMG,
     *               NLPRT, DIVSPC, MNLPG, MARG, NLINE, NSUM1, NSUM2,
     *               MT1, MT2, NPAGE, NCENPG
      COMMON /C2000/ KR, KW, IN, OUT
      DATA FT0/1H(/, FT1/1H /, FT2/1HX/, FT3/6H,1H-,I/, FT4/1H /,
     *     FT5/5H,1H-)/, ONE/1H1/, TWO/1H2/, THREE/1H3/
      IF(NLINE.GE.MXLINE) GO TO 11
      NLINE=NLINE+1
      IF(NLINE.GE.MXLINE) GO TO 11
      DO 10 I=NLINE,MXLINE
      WRITE(KW,5)
    5 FORMAT(1X)
   10 CONTINUE
   11 CONTINUE
      NPAGE=NPAGE+1
      IF(PGNUM.NE.1) GO TO 40
      IF(MT1.LE.0) GO TO 20
      DO 19 I=1,MT1
   19 WRITE(KW,5)
   20 CONTINUE
      NCENPG=NCOL/2+LFTMG-1
      IF(NCENPG.LT.10) ENCODE(1,21,FT1) NCENPG
   21 FORMAT(I1)
      IF(NCENPG.GE.10.AND.NCENPG.LE.99) ENCODE(2,22,FT1) NCENPG
   22 FORMAT(I2)
      IF(NPAGE.LT.10) FT4=ONE
      IF(NPAGE.GE.10.AND.NPAGE.LE.99) FT4=TWO
      IF(NPAGE.GE.100.AND.NPAGE.LE.999) FT4=THREE
      ENCODE(20,23,FORMT) FT0,FT1,FT2,FT3,FT4,FT5
   23 FORMAT(A1,A2,A2,A6,A1,A5)
      WRITE(KW,FORMT) NPAGE
C
      IF(MT2.LE.0) GO TO 31
      DO 30 I=1,MT2
   30 WRITE(KW,5)
   31 CONTINUE
      NLINE=TOPMG
      RETURN M
C
   40 IF(TOPMG.LE.0) GO TO 41
      DO 50 I=1,TOPMG
   50 WRITE(KW,5)
   41 NLINE=TOPMG
      RETURN M
      END
```

```
C*****************************    E N D C H K    *************************C
C                                                                          C
C     THIS SUBROUTINE CHECKS FOR "END" STATEMENT OF PROGRAM SECTIONS.      C
C                                                                          C
C*****************************************************************************C
      SUBROUTINE ENDCHK,RETURNS(N1,N2)
      REAL MARG(137)
      INTEGER TOPMG,BOTMG,PGNUM,DIVSPC,CHKFNP,OUT,ENDFLG
      DIMENSION END(3)
      COMMON /C1000/ CHKFNP, PGNUM, NCOL, LFTMG, MXLINE, TOPMG, BOTMG,
     *               NLPRT, DIVSPC, MNLPG, MARG, NLINE, NSUM1, NSUM2,
     *               MT1, MT2, NPAGE, NCENPG
      COMMON /C2000/ KR, KW, IN, OUT
      COMMON /C4000/ CARD(137),ENDFLG,NBCOL
      DATA BLANK/1H /, END/1HE, 1HN, 1HD/
      ENDFLG=0
      DO 10 I=1,6
      IF(CARD(I).NE.BLANK) RETURN N2
   10 CONTINUE
C
C     CHECK COL. 7 TO 72 FOR "END",ALL OTHER COLUMNS MUST BE BLANK
      NB=70
      IF(NBCOL.LT.70) NB=NBCOL
      ICOL=6
      DO 20 I=7,NB
      ICOL=ICOL+1
      IF(CARD(ICOL).EQ.BLANK) GO TO 20
      IF(ENDFLG.EQ.1) GO TO 50
C     CHECK FOR "END"
      IF(CARD(ICOL).NE.END(1).OR.
     *   CARD(ICOL+1).NE.END(2).OR.
     *   CARD(ICOL+2).NE.END(3))  GO TO 50
C     "END" LOCATED
      ENDFLG=1
      ICOL=ICOL+2
C     CONTINUE CHECKING FOR VALIDITY OF "END"
C
   20 CONTINUE
      IF(ENDFLG.NE.1) RETURN N2
C
C     VALID "END" STATEMENT LOCATED
      IF(NBCOL.EQ.0) NBCOL=1
      IF(LFTMG.GT.0) WRITE(KW,3) (MARG(I),I=1,LFTMG),(CARD(I),I=1,NBCOL)
      IF(LFTMG.LE.0) WRITE(KW,3) (CARD(I),I=1,NBCOL)
    3 FORMAT(137A1)
      NLINE=NLINE+1
      IF(NLINE.GE.NSUM1.OR.NLINE.GE.NSUM2) CALL NEWPAGE, RETURNS(45)
C
C     INSERT BLANK LINES BETWEEN SUBPROGRAMS
      IF(DIVSPC.LE.0) GO TO 45
      DO 40 I=1,DIVSPC
      IF(NLINE.GT.NSUM1) CALL NEWPAGE,RETURNS(45)
      WRITE(KW,4)
    4 FORMAT(1X)
   40 NLINE=NLINE+1
```

```
C
   45 ENDFLG=0
      RETURN N1
C
   50 ENDFLG=0
      RETURN N2
C
      END
```

Annexe D
Version structurée d'un tableau de données

DOCUMENTATION AND CALCULATIONS
****

RUNOFF MEASUREMENTS PINE POINT NEAR W17, OCT.19-22,1978

PROPELLOR DESCRIPTIONS FOR CURRENT METER USED:

PROPELLOR 1 (CODE: 19279-6) HAS EQUATIONS
V= 0.0955*N + 0.0600 FOR N < 3.4000
V= 0.1035*N + 0.0300 FOR N >= 3.4000

PROPELLOR 2 (CODE: 584-2) HAS EQUATIONS
V= 0.0983*N + 0.0460 FOR N < 4.4000
V= 0.1007*N + 0.0350 FOR N >= 4.4000

PROPELLOR 3 (CODE: 23295-3) HAS EQUATIONS
V= 0.2298*N + 0.0520 FOR N < 1.6000
V= 0.2570*N + 0.0100 FOR N >= 1.6000

EXPLANATION OF VARIABLES SHOWN IN TABLES:

CM=CENTIMETRES
DEPTH=DEPTH MEASURED FROM BOTTOM OF CREEK TO WATER SURFACE, LAST NUMBER IS TOTAL DEPTH OF CREEK
DISTANCE=DISTANCE MEASURED FROM LEFT TO RIGHT BANK, LOOKING UPSTREAM, FIRST DISTANCE IN CROSSECTION STANDS FOR LEFT BANK, LAST DISTANCE FOR RIGHT BANK
L=LITRES
M=METRES
N=NUMBER OF REVOLUTIONS/SECOND, AS MEASURED
PROP=PROPELLOR NUMBER
REV=REVOLUTIONS
S=SECONDS
V=VELOCITY IN METRES/SECOND

NUMBER OF RUNOFF STATIONS: 3

1.DISCHG.CHAN. W17, .5 MI. SOUTH OF RESOLUTION HIGHWAY, OCT.19/78 T:14.45

NUMBER OF MEASUREMENT PROFILES: 11

CROSSECTION OF THE CREEK:

LEFT TO RIGHT BANK IN CM, LOOKING UPSTREAM, TOTAL WIDTH = 265.0 CM

DISTANCE	(CM)	75.0	90.0	110.0	130.0	150.0	165.0	185.0
DEPTH	(CM)	26.0	31.0	50.0	63.0	78.0	106.0	137.0
DISTANCE	(CM)	205.0	230.0	260.0	270.0	300.0	330.0	340.0
DEPTH	(CM)	139.0	124.0	110.0	76.0	68.0	35.0	30.0

MEASUREMENTS AND RESULTS:

DISTANCE CM	DEPTH CM	REV	SECONDS	REV/S	PROP	VELOCITY CM/S
90.0	5.0	20.	25.0	0.80	3	23.6
	15.0	25.	25.0	1.00	3	28.2
	25.0	29.	25.0	1.16	3	31.9
	31.0	--VELOCITY AREA IS 0.08386 M**2/S--				
110.0	5.0	23.	25.0	0.92	3	26.3
	15.0	34.	25.0	1.36	3	36.5
	25.0	36.	25.0	1.44	3	38.3
	35.0	43.	25.0	1.72	3	45.2
	45.0	42.	25.0	1.68	3	44.2
	50.0	--VELOCITY AREA IS 0.18704 M**2/S--				
130.0	5.0	27.	25.0	1.08	3	30.0
	15.0	42.	25.0	1.68	3	44.2
	25.0	47.	25.0	1.88	3	49.3
	35.0	52.	25.0	2.08	3	54.5
	45.0	51.	25.0	2.04	3	53.4
	55.0	51.	25.0	2.04	3	53.4
	63.0	--VELOCITY AREA IS 0.29710 M**2/S--				
150.0	5.0	37.	25.0	1.48	3	39.2
	15.0	50.	25.0	2.00	3	52.4
	25.0	56.	25.0	2.24	3	58.6
	40.0	59.	25.0	2.36	3	61.7
	55.0	56.	25.0	2.24	3	58.6
	65.0	53.	25.0	2.12	3	55.5
	75.0	49.	25.0	1.96	3	51.4
	78.0	--VELOCITY AREA IS 0.42200 M**2/S--				
165.0	10.0	30.	25.0	1.20	3	32.8
	30.0	60.	25.0	2.40	3	62.7
	50.0	67.	25.0	2.68	3	69.9
	70.0	62.	25.0	2.48	3	64.7
	90.0	53.	25.0	2.12	3	55.5
	101.0	50.	25.0	2.00	3	52.4

	106.0	--VELOCITY AREA IS 0.59261 M**2/S--				
185.0	10.0	33.	25.0	1.32	3	35.5
	30.0	55.	25.0	2.20	3	57.5
	50.0	67.	25.0	2.68	3	69.9
	70.0	69.	25.0	2.76	3	71.9
	90.0	67.	25.0	2.68	3	69.9
	110.0	58.	25.0	2.32	3	60.6
	130.0	52.	25.0	2.08	3	54.5
	137.0	--VELOCITY AREA IS 0.81370 M**2/S--				
205.0	10.0	37.	25.0	1.48	3	39.2
	30.0	51.	25.0	2.04	3	53.4
	50.0	61.	25.0	2.44	3	63.7
	70.0	69.	25.0	2.76	3	71.9
	90.0	68.	25.0	2.72	3	70.9
	110.0	65.	25.0	2.60	3	67.8
	130.0	61.	25.0	2.44	3	63.7
	139.0	--VELOCITY AREA IS 0.84441 M**2/S--				
230.0	10.0	41.	25.0	1.64	3	43.1
	30.0	40.	25.0	1.60	3	42.1
	50.0	54.	25.0	2.16	3	56.5
	70.0	62.	25.0	2.48	3	64.7
	90.0	61.	25.0	2.44	3	63.7
	110.0	61.	25.0	2.44	3	63.7
	120.0	62.	25.0	2.48	3	64.7
	124.0	--VELOCITY AREA IS 0.68349 M**2/S--				
270.0	10.0	31.	25.0	1.24	3	33.7
	30.0	62.	25.0	2.48	3	64.7
	50.0	60.	25.0	2.40	3	62.7
	70.0	57.	25.0	2.28	3	59.6
	76.0	--VELOCITY AREA IS 0.40887 M**2/S--				
300.0	10.0	15.	25.0	0.60	3	19.0
	25.0	22.	25.0	0.88	3	25.4
	40.0	48.	25.0	1.92	3	50.3
	50.0	51.	25.0	2.04	3	53.4
	60.0	49.	25.0	1.96	3	51.4
	68.6	--VELOCITY AREA IS 0.24910 M**2/S--				
330.0	10.0	24.	25.0	0.96	3	27.3
	15.0	37.	25.0	1.48	3	39.2
	30.0	35.	25.0	1.40	3	37.4
	35.0	--VELOCITY AREA IS 0.10963 M**2/S--				

THE RUNOFF IS 1103.7 L/S .

---

2. DISCHARGE CHANNEL W17, AT RESOLUTION HIGHWAY OCT.19/78 T:17.00

NUMBER OF MEASUREMENT PROFILES: 13

CROSSECTION OF THE CREEK:

LEFT TO RIGHT BANK IN CM, LOOKING UPSTREAM, TOTAL WIDTH = 420.0 CM

DISTANCE (CM)	20.0	40.0	70.0	100.0	130.0	160.0	190.0
DEPTH (CM)	0.0	21.0	38.0	53.0	71.0	85.0	90.0
DISTANCE (CM)	220.0	250.0	280.0	310.0	340.0	370.0	400.0
DEPTH (CM)	93.0	92.0	86.0	80.0	66.0	46.0	32.0
DISTANCE (CM)	440.0						
DEPTH (CM)	0.0						

MEASUREMENTS AND RESULTS:

DISTANCE CM	DEPTH CM	REV	SECONDS	REV/S	PROP	VELOCITY CM/S
40.0	5.0	21.	25.0	0.84	3	24.5
	11.0	32.	25.0	1.28	3	34.6
	17.0	27.	25.0	1.08	3	30.0
	21.0	--VELOCITY AREA IS 0.05771 M**2/S--				
70.0	5.0	18.	25.0	0.72	3	21.7
	15.0	31.	25.0	1.24	3	33.7
	25.0	34.	25.0	1.36	3	36.5
	35.0	38.	25.0	1.52	3	40.1
	38.0	--VELOCITY AREA IS 0.12128 M**2/S--				
100.0	5.0	19.	25.0	0.76	3	22.7
	15.0	28.	25.0	1.12	3	30.9
	25.0	34.	25.0	1.36	3	36.5
	35.0	38.	25.0	1.52	3	40.1
	45.0	41.	25.0	1.64	3	43.1
	53.0	--VELOCITY AREA IS 0.18344 M**2/S--				
130.0	5.0	26.	25.0	1.04	3	29.1
	15.0	39.	25.0	1.56	3	41.0
	25.0	36.	25.0	1.44	3	38.3
	40.0	45.	25.0	1.80	3	47.3
	55.0	48.	25.0	1.92	3	50.3
	67.0	48.	25.0	1.92	3	50.3
	71.0	--VELOCITY AREA IS 0.30357 M**2/S--				
160.0	5.0	29.	25.0	1.16	3	31.9
	15.0	39.	25.0	1.56	3	41.0
	25.0	46.	25.0	1.84	3	48.3
	40.0	47.	25.0	1.88	3	49.3
	55.0	55.	25.0	2.20	3	57.5
	70.0	54.	25.0	2.16	3	56.5

	80.0	55.	25.0	2.20	3	57.5
	85.0	--VELOCITY AREA IS 0.41775 M**2/S--				
190.0	5.0	28.	25.0	1.12	3	30.9
	15.0	42.	25.0	1.68	3	44.2
	25.0	44.	25.0	1.76	3	46.2
	40.0	54.	25.0	2.16	3	56.5
	55.0	51.	25.0	2.04	3	53.4
	70.0	62.	25.0	2.48	3	64.7
	85.0	59.	25.0	2.36	3	61.7
	90.0	--VELOCITY AREA IS 0.46786 M**2/S--				
220.0	5.0	34.	25.0	1.36	3	36.5
	15.0	44.	25.0	1.76	3	46.2
	25.0	43.	25.0	1.72	3	45.2
	40.0	48.	25.0	1.92	3	50.3
	55.0	52.	25.0	2.08	3	54.5
	70.0	59.	25.0	2.36	3	61.7
	85.0	60.	25.0	2.40	3	62.7
	93.0	--VELOCITY AREA IS 0.48147 M**2/S--				
250.0	5.0	30.	25.0	1.20	3	32.8
	15.0	35.	25.0	1.40	3	37.4
	25.0	38.	25.0	1.52	3	40.1
	40.0	47.	25.0	1.88	3	49.3
	55.0	50.	25.0	2.00	3	52.4
	70.0	51.	25.0	2.04	3	53.4
	85.0	59.	25.0	2.36	3	61.7
	92.0	--VELOCITY AREA IS 0.43832 M**2/S--				
280.0	5.0	39.	25.0	1.56	3	41.0
	15.0	36.	25.0	1.44	3	38.3
	25.0	42.	25.0	1.68	3	44.2
	40.0	45.	25.0	1.80	3	47.3
	55.0	52.	25.0	2.08	3	54.5
	70.0	53.	25.0	2.12	3	55.5
	81.0	57.	25.0	2.28	3	59.6
	86.0	--VELOCITY AREA IS 0.41671 M**2/S--				
310.0	5.0	25.	25.0	1.00	3	28.2
	15.0	32.	25.0	1.28	3	34.6
	25.0	39.	25.0	1.56	3	41.0
	40.0	46.	25.0	1.84	3	48.3
	55.0	43.	25.0	1.72	3	45.2
	65.0	48.	25.0	1.92	3	50.3
	75.0	53.	25.0	2.12	3	55.5
	80.0	--VELOCITY AREA IS 0.34535 M**2/S--				
340.0	5.0	11.	25.0	0.44	3	15.3
	15.0	31.	25.0	1.24	3	33.7
	25.0	36.	25.0	1.44	3	38.3
	40.0	41.	25.0	1.64	3	43.1
	51.0	43.	25.0	1.72	3	45.2
	62.0	47.	25.0	1.88	3	49.3
	66.0	--VELOCITY AREA IS 0.24724 M**2/S--				
370.0	5.0	23.	25.0	0.92	3	26.3
	15.0	36.	25.0	1.44	3	38.3
	25.0	42.	25.0	1.68	3	44.2
	33.0	43.	25.0	1.72	3	45.2
	42.0	43.	25.0	1.72	3	45.2
	46.0	--VELOCITY AREA IS 0.17795 M**2/S--				

400.0	5.0	22.	25.0	0.88	3	25.4
	15.0	28.	25.0	1.12	3	30.9
	27.0	30.	25.0	1.20	3	32.8
	32.0	--VELOCITY AREA IS 0.09233 M**2/S--				

THE RUNOFF IS 1127.0 L/S .

3.DISCHG.CHAN. W17, UPSTREAM OF CONFLUENCE X15 CHANNEL, OCT.20/78 T:13.15

NUMBER OF MEASUREMENT PROFILES: 13

CROSSECTION OF THE CREEK:

LEFT TO RIGHT BANK IN CM, LOOKING UPSTREAM, TOTAL WIDTH = 440.0 CM

DISTANCE	(CM)	90.0	110.0	140.0	170.0	200.0	230.0	260.0
DEPTH	(CM)	0.0	20.0	32.0	45.0	53.0	65.0	67.0
DISTANCE	(CM)	290.0	320.0	350.0	380.0	410.0	460.0	490.0
DEPTH	(CM)	69.0	72.0	68.0	55.0	57.0	33.0	25.0
DISTANCE	(CM)	530.0						
DEPTH	(CM)	0.0						

MEASUREMENTS AND RESULTS:

DISTANCE CM	DEPTH CM	REV	SECONDS	REV/S	PROP	VELOCITY CM/S
110.0	5.0	25.	25.0	1.00	3	28.2
	10.0	31.	25.0	1.24	3	33.7
	15.0	35.	25.0	1.40	3	37.4
	20.0	--VELOCITY AREA IS 0.06245 M**2/S--				
140.0	5.0	14.	25.0	0.56	3	18.1
	15.0	32.	25.0	1.28	3	34.6
	25.0	39.	25.0	1.56	3	41.0
	32.0	--VELOCITY AREA IS 0.09968 M**2/S--				
170.0	5.0	21.	25.0	0.84	3	24.5
	15.0	36.	25.0	1.44	3	38.3
	25.0	40.	25.0	1.60	3	42.1
	35.0	43.	25.0	1.72	3	45.2
	41.0	45.	25.0	1.80	3	47.3
	45.0	--VELOCITY AREA IS 0.17110 M**2/S--				
200.0	5.0	28.	25.0	1.12	3	30.9
	15.0	42.	25.0	1.68	3	44.2
	25.0	51.	25.0	2.04	3	53.4
	35.0	50.	25.0	2.00	3	52.4
	45.0	48.	25.0	1.92	3	50.3
	53.0	--VELOCITY AREA IS 0.24186 M**2/S--				
230.0	5.0	22.	25.0	0.88	3	25.4
	15.0	41.	25.0	1.64	3	43.1
	25.0	51.	25.0	2.04	3	53.4
	35.0	57.	25.0	2.28	3	59.6
	45.0	56.	25.0	2.24	3	58.6
	60.0	51.	25.0	2.04	3	53.4
	65.0	--VELOCITY AREA IS 0.31798 M**2/S--				
260.0	5.0	17.	25.0	0.68	3	20.8

	15.0	54.	25.0	2.16	3	56.5
	25.0	58.	25.0	2.32	3	60.6
	40.0	60.	25.0	2.40	3	62.7
	50.0	57.	25.0	2.28	3	59.6
	61.0	53.	25.0	2.12	3	55.5
	67.0	--VELOCITY AREA IS 0.35327 M**2/S--				
290.0	5.0	40.	25.0	1.60	3	42.1
	15.0	59.	25.0	2.36	3	61.7
	25.0	63.	25.0	2.52	3	65.8
	40.0	65.	25.0	2.60	3	67.8
	55.0	63.	25.0	2.52	3	65.8
	65.0	59.	25.0	2.36	3	61.7
	69.0	--VELOCITY AREA IS 0.41980 M**2/S--				
320.0	5.0	31.	25.0	1.24	3	33.7
	15.0	55.	25.0	2.20	3	57.5
	25.0	62.	25.0	2.48	3	64.7
	40.0	69.	25.0	2.76	3	71.9
	55.0	69.	25.0	2.76	3	71.9
	68.0	71.	25.0	2.84	3	74.0
	72.0	--VELOCITY AREA IS 0.45423 M**2/S--				
350.0	5.0	24.	25.0	0.96	3	27.3
	15.0	50.	25.0	2.00	3	52.4
	25.0	59.	25.0	2.36	3	61.7
	40.0	70.	25.0	2.80	3	73.0
	52.0	70.	25.0	2.80	3	73.0
	64.0	71.	25.0	2.84	3	74.0
	68.0	--VELOCITY AREA IS 0.41335 M**2/S--				
380.0	5.0	37.	25.0	1.48	3	39.2
	15.0	52.	25.0	2.08	3	54.5
	25.0	62.	25.0	2.48	3	64.7
	35.0	69.	25.0	2.76	3	71.9
	45.0	75.	25.0	3.00	3	78.1
	51.0	74.	25.0	2.96	3	77.1
	55.0	--VELOCITY AREA IS 0.34173 M**2/S--				
410.0	5.0	24.	25.0	0.96	3	27.3
	15.0	38.	25.0	1.52	3	40.1
	25.0	55.	25.0	2.20	3	57.5
	35.0	66.	25.0	2.64	3	68.8
	45.0	67.	25.0	2.68	3	69.9
	53.0	74.	25.0	2.96	3	77.1
	57.0	--VELOCITY AREA IS 0.31492 M**2/S--				
460.0	5.0	27.	25.0	1.08	3	30.0
	15.0	46.	25.0	1.84	3	48.3
	23.0	46.	25.0	1.84	3	48.3
	30.0	44.	25.0	1.76	3	46.2
	33.0	--VELOCITY AREA IS 0.13586 M**2/S--				
490.0	5.0	27.	25.0	1.08	3	30.0
	13.0	38.	25.0	1.52	3	40.1
	20.0	36.	25.0	1.44	3	38.3
	25.0	--VELOCITY AREA IS 0.08558 M**2/S--				

THE RUNOFF IS 1069.8 L/S .

SUMMARY OF RESULTS
****

RUNOFF MEASUREMENTS PINE POINT NEAR W17, OCT 19-22,1978

1.DISCHG.CHAN. W17, .5 MI. SOUTH OF RESOLUTION HIGHWAY, OCT.19/78 T:14.45
*** THE RUNOFF IS 1103.7 L/S ***

2 DISCHARGE CHANNEL W17, AT RESOLUTION HIGHWAY OCT.19/78 T:17.00
*** THE RUNOFF IS 1127.0 L/S ***

3.DISCHG.CHAN. W17, UPSTREAM OF CONFLUENCE X15 CHANNEL, OCT.20/78 T:13.15
*** THE RUNOFF IS 1069.8 L/S ***

Annexe E
Procédure PLAY

```
PLAY(COMP=1,COMLIST=0,BINARY=1,INFILE=1,OUTFILE=1)
RFL,50000.
GRAB,COMMENT.
IF(NUM(INFILE))GOTO,9A.
IF(NUM(OUTFILE))GOTO,9B.
IF(NUM(COMP))GOTO,2A.
GET,COMP,INFILE.
REWIND,LGO,COMP,COMLIST,INFILE.
FTN,I=COMP,L=COMLIST.
COMMENT. PROGRAM HAS BEEN COMPILED.
IF(NUM(COMLIST))GOTO,1A.
REPLACE,COMLIST.
COMMENT. COMPILER LISTING HAS BEEN SAVED.
1A,IF(NUM(BINARY))GOTO,1B.
REPLACE,LGO=BINARY.
COMMENT. BINARY PROGRAM HAS BEEN SAVED.
1B,REWIND,INFILE,OUTFILE,LGO.
LGO,,,INFILE,OUTFILE.
REWIND,OUTFILE.
REPLACE,OUTFILE.
COMMENT. OUTPUT FILE HAS BEEN CREATED AND SAVED.
GOTO,9E.
2A,IF(NUM(BINARY))GOTO,9C.
GET,BINARY,INFILE.
COMMENT. BINARY PROGRAM IS AVAILABLE.
REWIND,BINARY,INFILE,OUTFILE.
BINARY,,,INFILE,OUTFILE.
REWIND,OUTFILE.
REPLACE,OUTFILE.
COMMENT. OUTPUT FILE HAS BEEN CREATED AND SAVED.
GOTO,9E.
9A,COMMENT. MISSING INPUT FILE.
GOTO,9E.
9B,COMMENT. MISSING OUTPUT FILE NAME.
GOTO,9E.
9C,COMMENT. MISSING PROGRAM FILE.
9E,RFL,6000.
COMMENT.            *** END OF RUN ***
*
```

UTILISATION DE LA PROCÉDURE PLAY

Pour lancer l'exécution des ordres de gestion de la procédure PLAY, l'utilisateur doit introduire l'une des instructions suivantes:

```
CALL,PLAY(COMP=N1,COMLIST=N2,BINARY=N3,
INFILE=N4,OUTFILE=N5)
```

ou

```
-PLAY(COMP=N1,COMLIST=N2,BINARY=N3,
INFILE=N4,OUTFILE=N5)
```

où N1 = nom du fichier qui contient le programme LAYOUT,
N2 = nom du fichier qui contiendra le listage de la compilation du programme LAYOUT, y compris les instructions du programme et celles du compilateur,
N3 = nom du fichier qui contiendra la version binaire du programme LAYOUT,
N4 = nom du fichier à structurer, et
N5 = nom du fichier qui contiendra la sortie structurée.

Il n'est pas nécessaire de fournir tous les paramètres des fichiers quand on appelle PLAY. Les noms de fichiers pour les paramètres INFILE et OUTFILE sont obligatoires. S'ils ne sont pas présents, des messages d'erreur sont imprimés par la procédure PLAY et l'exécution de celle-ci est interrompue. Il faut aussi affecter un nom de fichier soit à COMP, soit à BINARY. S'il y a une erreur, la procédure PLAY n'est pas exécutée; ce sont plutôt les messages d'erreur mentionnés au tableau 3 qui sont imprimés.

Le fichier COMLIST n'est pas employé au cours du déroulement du programme LAYOUT. Il sert d'intermédiaire pour mémoriser et retrouver le listage de la compilation du programme LAYOUT. Ce listage comprend les instructions du programme et les instructions normales du compilateur. Pour obtenir des instructions plus détaillées du compilateur, il faut modifier les instructions de compilation de la ligne 9 du listage de la procédure PLAY.

Les deux cas suivants illustrent et expliquent les affectations de fichiers les plus courantes pour la procédure PLAY.

Cas 1

```
CALL,PLAY(COMP=LAYOUT,COMLIST=FILE1,
BINARY=BINLAY,INFILE=FILE2,OUTFILE=FILE3)
```

Dans ce cas, l'instruction COMP=LAYOUT provoque la compilation du programme LAYOUT et l'instruction COMLIST=FILE1 provoque la mémorisation du listage de la compilation dans le fichier FILE1. La version binaire compilée du programme LAYOUT nécessaire pour les exécutions ultérieures de la procédure est mémorisée dans le fichier BINLAY. Ceci rend toute recompilation inutile pour les passages-machine ultérieurs. Le fichier FILE2 est celui qui doit être structuré par le programme LAYOUT. Le fichier FILE3 contiendra, après l'exécution du programme, la version finale structurée du fichier FILE2, prête pour impression au terminal. Si COMLIST=FILE1 est omis, aucun listage de la compilation ne sera mémorisé.

Cas 2

```
CALL,PLAY(BINARY=BINLAY,INFILE=FILE2,
OUTFILE=FILE3)
```

Dans ce cas, une version binaire du programme LAYOUT a été mémorisée dans le fichier BINLAY au cours d'un passage-machine antérieur. Il n'est donc pas nécessaire de compiler le programme LAYOUT, on peut omettre COMP et on doit omettre COMLIST. Comme pour le cas 1, FILE2 et FILE3 sont les fichiers entrée et sortie respectivement et ils doivent être inclus.